BOOTSTRAPPING CONVERSATIONAL SPEECH RECOGNITION SYSTEM USING NEURAL MACHINE TRANSLATION

Surabhi Punjabi, Harish Arsikere, Sri Garimella

Alexa Machine Learning, Amazon, Bangalore, India
{spunjabi, arsikere, srigar}@amazon.com

ABSTRACT

Building a conversational speech recognition system for a new language is constrained by the availability of interaction style utterances. Data collection is often expensive and limited by the speed of manual transcription. In this work, we advocate the use of neural machine translation as a data augmentation technique for bootstrapping language models in factored speech recognition systems. Translation offers a systematic way to incorporate live collections from the mature, resource-rich languages. However, the strategy of ingesting raw translations from a general purpose MT system is not effective owing to the presence of named entities, intra sentential code-switching and the domain mismatch between conversational data being translated and the parallel text used for training translation system. We explore sentence embeddings based data selection and model fine tuning for adaptation. We derive guidance from in-domain data by rescoring beams and filtering translations. A combination of these techniques yields a relative word error rate reduction of 7.8-15.6 % depending on the bootstrapping phase. Fine grained analysis reveals that translation aids the underrepresented interaction categories in particular. Experimental evidence establishes the efficacy of translation for supplementing transcribed collections, a strategy which could be instrumental for rapid language expansion.

Index Terms— speech recognition, neural machine translation, domain adaptation

1. INTRODUCTION

Bootstrapping an automatic speech recognition(ASR) system for a new language involves significant data collection and transcription overhead. For factored ASR systems, where the acoustic model (AM) and language model (LM) are trained independently, LM can be trained with additional unpaired text-only corpora to boost performance. This is especially helpful during the initial stages of model development. For a new language the typical supplemental LM resources include Wikipedia, news portals, blogs etc., which can be incorporated along with the limited transcribed data to circumvent the issue of cold start. However, for conversational agents like Alexa, Siri the utterances are usually short, goal directed and contain a lot of named-entities like SongName, ArtistName etc. (ex: *Play Moonlight Sonata by Beethoven*). This informal interaction style, characteristic of conversational data is absent in the online text sources, rendering them less effective for this task. As a result, the LM building relies mostly on utterance data and is limited by the speed of manual transcriptions and annotations.

There has been a growing interest in the area of data augmentation for ASR language modeling. [1] proposed the use of RNNLMs trained over the available transcripts, to generate synthetic samples for LM training. SeqGAN, a generative adversarial model for sequences has been recently employed for pretraining a code switched LM [2]. However, a precondition for successful generalization of these neural generative models is the presence of a substantial amount of in-domain utterance text, which is itself the bottleneck during bootstrapping phase.

Live utterances from mature languages such as English are a rich source of information. They are both *in-domain*, since they capture actual user interaction patterns of varying complexity, and *large-scale* owing to prolonged usage. Translation offers an elegant and cost-effective solution to leverage this existing data. Devising techniques for systematically incorporating translated data can be instrumental to achieving the goal of rapid language expansion for ASR. In this work we explore the efficacy and challenges associated with machine translation for the task of LM data augmentation.

The area of machine translation has witnessed sustained research efforts from both academia and industry [3, 4, 5]. It is also amongst the first success stories of the end-to-end neural paradigm for sequence modeling problems. Conventional phrase based statistical machine translation (PBSMT) [6] has shown to be been outperformed by attention based encoder-decoder recurrent neural networks [5] and transformer networks comprising of self-attention and feed forward network blocks [7].

Use of statistical machine translation (SMT) for data augmentation for keyword spotting [1] and ASR [8, 9, 10] has been explored in past. These works focus on incorporating raw translation output as a component in the interpolated n-gram language model. Our initial experiments with conver-

sational data however indicated that directly ingesting translations generated from off-the-shelf MT models yields a very high perplexity LM component. The primary reason for this is the domain mismatch between the training data for the MT model comprising of parallel text from web sources and the informal style interaction data used for translation. This observation of MT output being sensitive to mismatch in training and inference data distribution is consistent with the findings of some of the previous works on MT adaptation [11].

Recent work by [12] investigates the use of translation for bootstrapping NLP systems. They employ SMT system for generating initial translations and use the alignments between source and target to retain and re-sample slot entities. This approach addresses the problem of named entities conversion, yet the bigger issue of domain mismatch still remains open.

In this work, we explore the synergies between neural machine translation and speech recognition for data augmentation. We work towards bootstrapping ASR system for Hindi language. Along with the limited availability of representative transcribed data, an additional challenge in this setting is that of code switching. In typical Hindi utterances people often code mix with English within a single sentence.

We evaluate different architectures for building EN $\rightarrow$ HI translation models. We then elaborate on the pitfalls associated with using off-the-shelf translations. We observe some initial gains by inferring alignments from attention weights, an approach that enables us to preserve and resample the named entities. We build on this approach to introduce code switching in the translated data and handle transliteration.

We then delve into the deeper issue of domain inconsistency. To this end, we develop a data selection strategy for MT model training based on in-domain similarity. This is an extension of [13] for the fully unsupervised setting. We also assess the model fine tuning approach by adding parallel in-domain synthetic pairs. For further adaptation, we incorporate guidance from a statistical model built using transcribed data by rescoring the decoded translation beams. Finally, we evaluate different quality metrics for retaining only the high quality translations for building the final translation component. Along with exploring the confluence of NMT and ASR, this investigation is the first to probe into domain adaptation for MT for conversational speech recognition.

Using a combination of the adaptation techniques, we obtain relative word error rate reduction (WERR) ranging from 7.8-15.6 $\%$ by supplementing LM with translation data w.r.t. transcribed data-only baselines. The gains are pronounced for categories like *Knowledge, Shopping* which were originally under-represented in the transcribed collections. The experimental evidence suggests that translation is a promising augmentation technique which can be instrumental in alleviating the prohibitively high data requirements for data collection during bootstrapping.

2. MACHINE TRANSLATION FOR DATA AUGMENTATION

2.1. Building translation model

Neural machine translation has emerged as the dominant paradigm for MT research. We assess the popular neural architectures for the task of building EN$\rightarrow$HI translation model. Sequence-to-sequence framework with attention, proposed in [4] comprises of two recurrent neural networks: *Encoder* which reads the source sentence tokens $(x_1, x_2, ..x_t)$ to generate continuous representations $(h_1, h_2, ..h_t)$, and *Decoder* which outputs symbols $(y_1, y_2...y_n)$, conditioned on the previous outputs as well as a context vector **c**, derived as the weighted sum of the encoder hidden states **h**. Attention mechanism serves as an implicit alignment model, allowing the decoder to focus on relevant source segments. As we shall see, it is this alignment property of these recurrent models which can be a key enabler in generating meaningful translations from named-entity heavy utterances. Transformer networks proposed in [7] eliminate recurrence in favor of parallelism and rely solely on attention. The encoder and decoder comprise of stacked self-attention and fully connected layers. Transformer networks represent the current state-of-the-art for NMT.

For training the translation models, we procure a corpora of $\sim$ 8.4M parallel (EN, HI) sentence pairs by crawling different web sources. We employ BLEU (Bilingual Evaluation Understudy) score on the validation set created from the same data to assess the translation quality. Figure 1 captures the performance of these models with different model configurations. Here T indicates transformer architecture, R implies recurrent. The best performing encoder-decoder with attention model achieves a BLEU score of 43.8 as compared to 46.4 achieved by the transformer architecture.

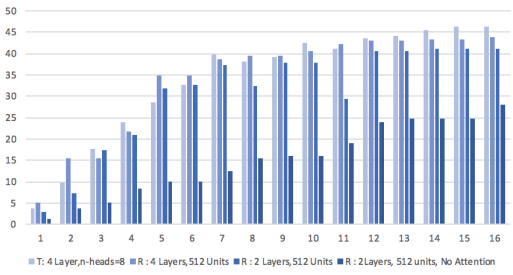

Fig. 1. *BLEU score over epochs.*

2.2. Incorporating raw translations: An initial study

We use the trained translation models to transform user interaction sentences from English-US locale to Hindi. In the initial experiments, directly ingesting the raw translations resulted in an extremely high perplexity language model. Upon further analysis, we found three key explanatory factors for this. Firstly, typical user interactions with voice controlled

agents for ex: a song request, contain a lot of named entities. The general purpose EN→HI MT system generates translations for those entities as well which is not desirable. Second factor is the absence of code switching in translations, which are purely in Hindi, owing to the nature of training data. However, our empirical analysis reveals a dominant trend of code mixing with English within a single sentence. Given the extent of intra-sentential code mixing, it seems imperative for the translations to capture it as well for it to add value to the downstream language modeling task.

Finally, the most subtle nuance is that the out-of-domain nature of MT training data (news items, wiki articles, etc.) leads to a lack of informal interaction style in the generated translations, an inherent attribute of the user-device interaction. This domain mismatch issue has been observed in other machine translation settings as well. We provide some anecdotal examples in table 1 to illustrate these factors.

Utterance	Raw Translation
read a sample of wings of fire book	आग किताब के पंख का एक नमूना पढ़ें
play barry white radio on iheartradio	कैटवॉक पर बैरी व्हाइट रेडियो खेलते हैं
purchase tickets to go guardians of the galaxy	आकाशगंगा के अभिभावकों को जाने के लिए टिकट खरीद
play a sample from harry potter book	हैरी पॉटर पुस्तक से एक नमूना खेलने

Fig. 2. *Examples of raw translation outputs.*

2.3. Post-editing translations

Identifying and post-processing the named entities in the output of statistical translation models is straightforward owing to the fact that source-target alignment model is learnt explicitly. The idea of preserving and resampling the named entities by combining this source-target alignment information and source token metadata has been explored in previous works [12].

In the case of NMT models, the separate components of the conventional SMT are folded into an all neural architecture. With this simplification however, assessing which source tokens is responsible for generating a target token becomes tricky.

The attention module in the encoder-decoder architecture performs implicit alignment to some extent. To understand this better, we can look at equation 1, where we notice that the attention weight $\alpha_{i,j}$ determines the weight assigned by decoder at time step j to encoder state i. This implies that attention weights can be used to approximate alignments in the encoder-decoder architecture.

The problem of deriving alignments aggravates for transformer networks with self attention and multiple attention heads. There has been some recent work for alleviating this issue by adding an additional alignment head to the base architecture [14]. Owing to the relative feasibility of alignment extraction, we make the modeling design decision of using the encoder-decoder networks with attention for our experiments.

With the choice of NMT architecture in place, we use the attention weights derived during decoding to post-edit translations. We consider the source token corresponding to encoder position with maximum attention weight to be *aligned* with the target being generated. Using the metadata in annotations (*SongName, ArtistName*) we identify named entities, and resample them with Hindi catalogs. This postprocessing step drastically brings down the perplexity of the translation component. On similar lines, we simulate code switching in the translations, by probabilistically copying-over source tokens. The probability of retention of an English token is directly proportional to the relative frequency in the in-house collection data.

2.4. Domain adaptation

We now attempt to address the more subtle challenge of domain mismatch. Most of the prominent works in MT model adaptation like backtranslation [15], shallow fusion [16] etc., assume the presence of large target side monolingual corpus for boosting the fluency of translations. This is a sharp contrast to our setting, where we can at most assume limited target in-domain data (transcribed collections) present at our disposal for adaptation. With this constraint in place, we experiment with three broad classes of techniques for adaptation: a) data selection for MT training, b) model fine tuning and rescoring, and c) post-processing and filtering translations.

2.4.1. Data selection for MT training

Selecting NMT training data sentences the out-of-domain parallel corpora which is similar to in-domain data has been explored in [13]. The central idea of that work is to train an NMT system using both in-domain and out-of-domain parallel corpora. Then the similarity between learnt encoder representations of the sentences is used to define their semantic closeness. Sentence selection then is based on the relative similarity to in-domain v/s out-domain centroid.

We adopt a similar data-centric approach for adaptation. In our setting however, in-domain parallel corpus is not available. Hence representing a sentence via NMT encoder embedding is not feasible. In order to filter the MT training samples, we need to resort to learning unsupervised embeddings to quantify closeness of the target side MT corpora with the in-domain transcriptions available.

For generating the distributed sentence representation in an unsupervised manner, we compare the following techniques : (i) unweighted averaging of word vectors, (ii) smooth inverse frequency [17] : deriving sentence embeddings as weighted average of word vectors followed by removal of the projection on the first singular vector and (iii) language agnostic sentence representations (LASER) [18]: an open

source pre-trained biLSTM encoder for generating multilingual sentence embeddings that generalize across language and NLP tasks. The first approach is appealing owing to its simplicity. In the second approach, taking word frequency into account is the demarcating factor. Potential cross-lingual generalization is the advantage offered by the third approach. We aim to investigate the relative merits of these approaches for the task of similarity search for domain adaptation for conversational speech.We use the pretrained FastText word embeddings [19] to represent the word vectors.

Using each of these approaches, we generate sentence embedding vectors for both in-domain corpus and out-of-domain target side sentences. Let the embeddings vector be denoted by $v_{in}(s)$ and $v_{out}(s)$. On similar lines as [13], the relative score for the sentence similarity is defined as [EQN] , where $C_{in}(s)$ and $C_{out}(s)$ are the average centroid vectors.

2.4.2. Model Finetuning

Backtranslation [15] is a popular approach for adaptation where a target to source translation model is trained to translate the target-side monolingual corpus and generate synthetic (source, target) pairs which can be leveraged for the original source to target model training.

Owing to the absence of a large target monolingual corpora, we resort to a different approach for synthetic corpus generation. We generate pseudo pairs by translating the source sentences using an initially trained NMT model, and performing post-editing to retain named entities. With this additional parallel data, we now fine tune the model for certain epochs. As we shall discuss in the experimental section, this type of fine tuning is susceptible to overfitting, and it is imperative to perform early stopping.

2.4.3. Rescoring with in-domain LM

In order to further boost the fluency of translations, we rescore the hypotheses obtained through beam search decoding with a statistical language model built from the in-domain data. More precisely, the score of a translation hypothesis is computed as a weighted sum of MT decoding score and the LM score. [Equation]. The choice of a statistical n-gram LM with back-off is motivated by its robustness under low-resource conditions as compared to RNNLM.

2.4.4. Filtering translations

As a final step, we attempt to filter out the spurious translations by applying threshold over some quality metric. The challenging task here is to define the "goodness" of a translated output. To this end, we explore the MT score, i.e. the product of conditional probability of the output tokens assigned by the decoder, as a candidate metric. We also evaluate the approach of generating scores from the LM built using transcribed data to assess the quality of translation data. Using these scores, we select the top-x percentile of the translation output.

3. RESULTS AND DISCUSSION

3.1. Experimental Setup

We conduct experiments using 180 hours (200K utterances) of Hindi-English code-switched speech. This dataset comprises of Hindi utterances collected using Cleo, a skill that enables users to teach local languages to voice assistants via prompts. These prompts cover use cases like request for songs or knowledge questions. These natural utterances represent the in-domain component in our experiments. We follow the factored ASR architecture, and the AM is a hybrid DNN-HMM model. Language model variants are built by learning n-grams (n=4) with Katz backoff on the training data.

We build the baseline LM using the in-domain transcriptions only. The augmentation dataset is procured by translating 9.8M English-US transcripts using the adapted NMT models followed by post-editing and filtering. For the evaluation candidates, the LM is built by interpolating the manually transcribed and machine translated component. The interpolation weights are typically learnt based on the perplexity on a hold-out in-domain dataset. We assign some floor interpolation weight to the translation component to ensure it receives sufficient representation in the LM in order to reliably study its impact on ASR performance.

3.2. Results

We summarize the experimental results in Table 1. The approach of ingesting raw translations yields a high perplexity augmentation component, resulting in a negative WERR. We observe consistent improvements by introducing attention-weight based post-editing. In particular, retaining the named entities alone brings down the perplexity significantly. This, coupled with resampling with local catalogues and code switching yields a big jump in performance, achieving 5.83% WERR. We observe a further boost in WERR by employing adaptation. We first study the impact of different approaches in isolation, before evaluating the consolidated effect. In all the following results, post-editing is applied in conjunction with the adaptation technique.

MT training data selection yields WERR improvement though BLEU score takes some hit owing to less parallel data. Amongst the sentence representation techniques, LASER and SIF embeddings outperform the unweighted averaging approach in terms of translation quality. This is reflected by the BLEU score attained after same number of training epochs. Interestingly, while the latter approach achieves lowest perplexity, the gains don't carry-over while measuring WERR. SIF embedding based selection achieves the highest WERR

Adaptation	Approach	PPL	Relative WERR %
No adaptation	Raw Translations	11941.08	-1.81
Post Editing	NE Copy-over	2889.45	2.36
	NE Resampling	1241.52	4.62
	Code mixing + NE Resampling	936.64	5.83
Data Selection	Unweighted avg. (BLEU : 29.065)	662.33	6.94
	SIF (BLEU : 37.805)	686.97	7.23
	LASER (BLEU : 37.381)	704.12	7.14
Rescoring	beam-size=5	792.92	6.28
	beam-size=20	852.16	5.88
Model Finetuning	n-epochs=3	726.62	6.84
	n-epochs=10	983.64	5.23
Filtering translations	MT score - top 85 %	1109.44	4.82
	MT score - top 75%	1327.56	3.37
	MT score - top 65%	1426.18	2.16
Filtering translations	SLM score- top 85%	793.73	6.33
	SLM score- top 75%	892.92	6.82
	SLM score- top 65%	878.16	5.94
Combined	(i) SIF Selection + Rescoring + SLM-top 85 %	584.24	7.59
	(i) + Model Fine tuning	564.06	7.84

Table 1. *Results with different NMT adaptation strategies for augmentation. PPL is evaluated on a hold-out in-domain dataset. Relative WERR measures WER reduction w.r.t baseline with no translation data.*

of 7.23%, followed closely by LASER encoder representation.

Rescoring the decoded beams using LM built over transcribed data yields a WERR of 6.28%. Increasing the beam width from 5 to 20 results in a drop in WERR, implying that the during decoding, the head portion of translation outputs contains the hypotheses that are helpful for ASR performance.

For the model *fine tuning* approach, number of additional training epochs is an important parameter. We observe a WERR of 6.84 % when the n-epochs for additional training is 3, as compared to 5.23% for 10 epochs. Increasing the number of passes on the synthetic data generated using an initially trained model perpetuates the effect of model reinforcing its own errors. This potential overfitting makes early stopping imperative for obtaining gains using pseudo-parallel data based tuning.

In the experiments focusing on *translation output filtering*, MT score did not turn out to be an effective metric for quality evaluation, indicated both by perplexity and WERR. We obtained interesting insights by ranking translations using in-domain LM scores. Moving from all translations, to top-85 % and then to top-75 % the WERR rises to 6.82 %. Making filtering conservative beyond this point degrades performance. An important caveat of the LM guided filtering approach is that the patterns which are underrepresented in the initial collections will receive low LM score. Since the transcribed volume is itself small, some of these patterns could have been but are complementary for the overall ASR performance. This could explain why pruning off only the low score translations manifests in WERR boost.

Combining the SIF selection, finetuning, rescoring and language model based filtering approach results in a WERR of 7.84 %.

3.3. Impact of in-domain dataset scale

We now attempt to address the following question: what are the relative gains provided by the translation data during different phases of bootstrapping? In particular, we measure the WERR between a baseline and translation-augmented language model, at varying levels of available transcribed utterances.

From table 3, we notice that the gains are pronounced, and go as high as 15.65 %, when the available transcribed data is low. The gains come down we keep increasing the in-domain data. Note that in these experiments, we use the same AM (trained on 180 hours of data), in order to precisely study the effect of data augmentation for LM. The WERR we report is hence an underestimate, and will be typically much higher, if the acoustic model was trained on similar levels of training data.

Transcribed Volume	WERR %
10K	15.65
20k	13.18
50k	9.42
100k	8.98
200k	7.84

Table 2. *WERR for varying levels of in-domain dataset.*

3.4. Impact of floor weight for interpolation

We now investigate the effect of changing the floor weight parameter, which in principle allows us to control the relative importance of the translated component v/s the in-domain component in the overall interpolated language model. This is important especially in the light of domain mismatch with the validation data.

Floor Weight	WERR %
0.1	5.78
0.15	7.04
0.25	7.84
0.3	7.49
0.35	7.24
0.4	6.58

Table 3. *WERR for varying floor interpolation weights for translation component.*

We notice WERR fluctuation by varying floor weights: a very low value renders translation component ineffective whereas the other extreme can undermine the transcription component. Floor weight sweep can thus provide empirical guidance for setting the floor weight at a given data level.

3.5. Impact on different interaction scenarios

The scenarios covered via Cleo prompts can span different use cases. In order to procure fine grained insights on the effect of translations, we manually bucketed some of the test utterances and studied WERR for each category.

A key observation from figure 3 is that use cases related to shopping and information which were not well-represented in the transcribed collections achieve higher WERR as compared to their popular counterparts such as song requests and reminders. This reinforces the fact that translations can effectively complement the transcribed data for different interaction use cases.

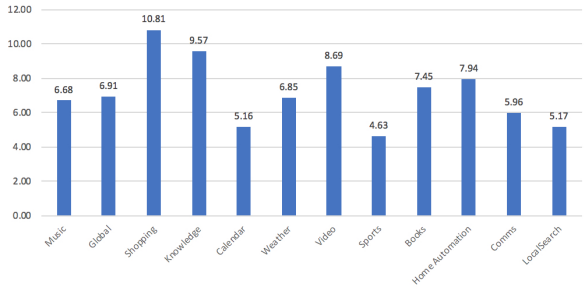

Fig. 3. *Relative WERR for different interaction use cases.*

4. CONCLUSION

Machine translation is a promising avenue pivotal to the rapid language expansion goal for ASR. In this work, we explored the challenges associated with neural machine translation and devised techniques for addressing issues with named entities, code switching and domain mismatch. Using a combination of sentence embedding based selection, model finetuning, rescoring beams, LM based filtering and post-editing, we achieved a relative WERR of 7.8% for 180 hours of transcribed data. We examined the performance trajectory along different phases of bootstrapping, and observed relative WERR of upto 15.6% when minimal transcribed data is available. While studying domain-level WERR, we derived some interesting insights elucidating the complementary nature of translations. Though Hindi dataset is used as an experimental testbed in this work, the insights presented can be leveraged for bootstrapping other languages as well. Exploring semi-supervised and unsupervised translation can be a promising direction especially for the low resource languages where sufficient parallel corpora is not available.

5. REFERENCES

[1] Arseniy Gorin, Rasa Lileikytė, Guangpu Huang, Lori Lamel, Jean-Luc Gauvain, and Antoine Laurent, "Language model data augmentation for keyword spotting in low-resourced training conditions," in *Interspeech 2016*, 2016, pp. 775–779.

[2] Saurabh Garg, Tanmay Parekh, and Preethi Jyothi, "Code-switched language models using dual RNNs and same-source pretraining," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, Oct.-Nov. 2018, pp. 3078–3083, Association for Computational Linguistics.

[3] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio, "On the properties of neural machine translation: Encoder–decoder approaches," in *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, Doha,

Qatar, Oct. 2014, pp. 103–111, Association for Computational Linguistics.

[4] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, "Neural machine translation by jointly learning to align and translate," *CoRR*, vol. abs/1409.0473, 2015.

[5] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean, "Google's neural machine translation system: Bridging the gap between human and machine translation," *CoRR*, vol. abs/1609.08144, 2016.

[6] Philipp Koehn, Franz Josef Och, and Daniel Marcu, "Statistical phrase-based translation," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, Stroudsburg, PA, USA, 2003, NAACL '03, pp. 48–54, Association for Computational Linguistics.

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017.

[8] ArnarThor Jensson, Koji Iwano, and Sadaoki Furui, "Language model adaptation using machine-translated text for resource-deficient languages," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2008, no. 1, pp. 573832, Jan 2009.

[9] Horia Cucu, Andi Buzo, Laurent Besacier, and Corneliu Burileanu, "Smt-based asr domain adaptation methods for under-resourced languages: Application to romanian," *Speech Commun.*, vol. 56, pp. 195–212, Jan. 2014.

[10] Thang Vu, Dau-Cheng Lyu, Jochen Weiner, Dominic Telaar, Tim Schlippe, Fabian Blaicher, Eng Chng, Tanja Schultz, and Haizhou Li, "A first speech recognition system for mandarin-english code-switch conversational speech," in *ICASSP*, 03 2012.

[11] Chenhui Chu and Rui Wang, "A survey of domain adaptation for neural machine translation," *CoRR*, vol. abs/1806.00258, 2018.

[12] Judith Gaspers, Penny Karanasou, and Rajen Chatterjee, "Selecting machine-translated data for quick bootstrapping of a natural language understanding system," *CoRR*, vol. abs/1805.09119, 2018.

[13] Rui Wang, Andrew Finch, Masao Utiyama, and Eiichiro Sumita, "Sentence embedding for neural machine translation domain adaptation," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Vancouver, Canada, July 2017, pp. 560–566, Association for Computational Linguistics.

[14] Tamer Alkhouli, Gabriel Bretschner, and Hermann Ney, "On the alignment problem in multi-head attention-based neural machine translation," in *WMT*, 2018.

[15] Rico Sennrich, Barry Haddow, and Alexandra Birch, "Improving neural machine translation models with monolingual data," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Berlin, Germany, Aug. 2016, pp. 86–96, Association for Computational Linguistics.

[16] Çaglar Gülçehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loïc Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, "On using monolingual corpora in neural machine translation," *CoRR*, vol. abs/1503.03535, 2015.

[17] Jonas Mueller and Aditya Thyagarajan, "Siamese recurrent architectures for learning sentence similarity," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*. 2016, AAAI'16, pp. 2786–2792, AAAI Press.

[18] Mikel Artetxe and Holger Schwenk, "Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond," *CoRR*, vol. abs/1812.10464, 2018.

[19] Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov, "Learning word vectors for 157 languages," in *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*, 2018.