Charith Muppidi & Harshith Nanda

ACL Paper Summary

**Authors**

The title of the paper is “Learning Language Specific Sub-network for Multilingual Machine Translation.” It was written by three ByteDance employees - Zehui Lin, Liwei Wu, and Mingxuan Wang University of California Santa Barbara professor Lei Li.

**The Problem**

To quote the paper, “Multilingual neural machine translation aims at learning a single translation model for multiple languages.“ So far, it has been successful for bilingual machine translation, and can be used to translate multiple source languages into multiple target languages in a single model.

However, it has several problems. Most notable among these is that in order to accommodate multiple languages being translated into multiple other languages, the NMT has to split for each translation which leads to a worse performance than a bilingual baseline.

**Prior Work**

A standard multilingual NMT model uses a shared encoder and a shared encoder for all languages. There is what is known as the transfer-interference trade-off with this standard model where we have to boost the performance of low resource languages while maintaining the performance of high resource languages. This trade was solved by assigning parts of the model, such as encoders, decoders, hidden states and embeds, to be language specific.

**Unique Contributions**

The solution the team proposes is the Language Specific Sub-network (LaSS) method which will give one sub-network to each language pair. Each sub-network will have their own language-specific parameters while also sharing parameters with other languages. This gives us the advantages of being parameter efficient, since no extra parameters have to be trained to model language-specific features, and parameter interference is reduced which improves capacity and performance. LaSS also gives the model the ability to adapt to new language pairs without dramatically decreasing performance of the old pairs. It is also really good at zero-shot translation which is translation of known languages that the model has never seen together at training time. For example, if it has trained on French and Chinese separately then it can train on French-to-Chinese and vice versa with little issues. LaSS has shown to consistently outperform the baseline on all language pairs.

**Evaluation**

They used benchmarks from the International Conference on Spoken Language Translation (IWSLT) and the Conference on Machine Translation (WMT). They collected eight English-centric language pairs for IWSLT and 18 language pairs from low-resource to rice resource for WMT. They apply byte pair encoding to preprocess multilingual sentences and apply over-sampling to balance out the training data distribution.

They use the bilingual evaluation understudy (BLEU) algorithm, which evaluates the quality of the machine-translated text by comparing the machine’s output to a human’s choice of words. They also use win ratio (WR) which divides the total number of correct guesses with the total number of incorrect guesses. For zero-shot translation they used translation-language accuracy which checks how accurate the translation was.

**Citations and Conclusion**

The employees of ByteDance - Zehui Lin, Liwei Wu, Mingxuan Wang have 127, 135, and 1750 citations respectively. Professor Lei Li has 7653 citations.

LaSS is important because there are a plethora of uses for it. One use is translating a famous work into multiple different languages at once, reducing the time it takes to make a language-specific translation. Another use is reviving an endangered/extinct language by translating it into a living language and vice-versa. A person may be curious as to how a word in their native language is pronounced or spelt in a different language. In a country with so many different native languages, like the Philippines, South Africa, and Italy, you can translate the local languages and lingua franca to make communication easier. Because Lass allows for so many different types of translations, it can revolutionize the way we translate text.