https://www.napizia.com
eryk@wdowiak.me

Eryk Wdowiak
10 May 2021

Sicilian Translator: A Recipe for Low-Resource NMT

With patience and dedication to a clear long-term vision, you can create amazing things.

So I've been steadily assembling a corpus of parallel text to create a machine translator for the Sicilian language. It now translates simple sentences fairly well. With a little more work, we will soon have a good-quality translator.

Sicilian is a good case study for several reasons. First, the language has been continuously recorded since the Sicilian School of Poets joined the imperial court of Frederick II in the 13th century.

And in our times, Arba Sicula has spent the past 40 years translating Sicilian literature into English (among its numerous activities to promote the Sicilian language). In the course of their work with the many dialects of Sicilian, they also established a "Standard Sicilian," which is what has enabled us to create a high-quality corpus of Sicilian-English parallel text.

High-quality parallel text is the necessary ingredient in any neural machine translation project. And recent advances in the field have made it possible to develop neural machine translators with limited amounts of parallel text.

With just 16,945 translated sentence pairs containing 266,514 Sicilian words and 269,153 English words, our *Tradutturi Sicilianu* achieved a BLEU score of 25.1 on English-to-Sicilian translation and 29.1 on Sicilian-to-English.

That's a good result for a small amount of parallel text. And you can always add more parallel text. Augmenting our dataset with backtranslations and multilingual translation further increased our BLEU scores to 35.1 on English-to-Sicilian and to 34.6 on Sicilian-to-English.

The traditional recommendation for languages without any parallel text has been to create a rules-based translator, using a framework like *Apertium*. But rules are difficult to write and, after a certain point, writing more rules won't improve translation quality very much.

But you can always find ways to create more parallel text. With patience and dedication, you can assemble tens of thousands of sentence pairs. Then assemble thousands more to further improve translation quality.

So the next section describes our data sources. The section on subword splitting explains how we prepare sentences for translation. Then our "recipe" describes our method of training a translator on little parallel text. Finally, the section on multilingual translation explains how adding Italian-English translations to our dataset enables translation with Italian and further improves translation quality. And the last section concludes.

Data Sources

When we first set out to create a machine translator for the Sicilian language, we thought that the limited number of parallel texts, the diversity of the Sicilian language and the diverse ways that the Sicilian language has been written would make it impossible to use statistical methods to create a machine translator.

Just a few years ago, Koehn and Knowles (2017) calculated learning curves for English-to-Spanish translation. At 377,000 words, the BLEU scores were 1.6 for neural machine translation, 16.4 for statistical machine translation and 21.8 for statistical with a big language model.

Recent advances in the field of neural machine translation allowed us to obtain better scores with half the number of words. But first we had to collect some Sicilian text.

Repositories of open-source parallel text, like the OPUS project, do not have any Sicilian language resources. There are no government documents, Wikipedia articles or movie subtitles that we can use as a source of parallel text. But good resources can be found elsewhere.

To seed this project, Arthur Dieli kindly provided 34 translations of Giuseppe Pitrè's *Sicilian Folk Tales* and lots of encouragement. And Arba Sicula, which has been translating Sicilian literature into English for over 40 years, contributed its bilingual journal of Sicilian history, language, literature, art, folklore and cuisine.

Just as importantly, Arba Sicula developed a standard form of the language, providing the consistency we need in a sea of orthographic and dialectical diversity.

Most of our data comes from *Arba Sicula* articles. Some parallel text comes from Dr. Dieli's translations of Pitrè's *Folk Tales*. And some comes from translations of the homework exercises in the *Mparamu lu sicilianu* (Cipolla, 2013) and *Introduction to Sicilian Grammar* (Bonner, 2001) textbooks.

Although it only makes up a small portion of the dataset, adding the textbook examples yielded large improvements in translation quality on a test set drawn only from *Arba Sicula* articles. Just as a grammar book helps a human learn in a systematic way, it also seems to help a machine learn in a systematic way.

Another (ironic) source of parallel text is monolingual text. Efforts to create neural machine translators for other low-resource languages often involve the back-translation method developed by Sennrich et al. (2015), in which monolingual, target-side text is used to supplement the available parallel text.

We may make more use this method in the future. So far we have not used it much because assembling Sicilian monolingual text requires almost as much time as assembling parallel text. Nonetheless, we also have some leftover unmatched text, which we can use for back-translation.

For example, to develop our English-to-Sicilian model, we could automatically translate Sicilian text into English to create a "synthetic dataset" of real Sicilian sentences and synthetic English sentences. Then we would train a new English-to-Sicilian model on the combination of the parallel and synthetic data.

And in general, you can always find ways to assemble more parallel text.

Subword Splitting

In a recent case study, Sennrich and Zhang (2019) develop a set of best practices for low-resource neural machine translation and show that those best practices can achieve better translation quality than phrase-based statistical machine translation in a 100,000 word dataset derived from the 2014 German-English IWSLT.

In their best practices, they suggest using a smaller neural network with fewer layers, smaller batch sizes and a larger the dropout parameter. And their largest improvements in translation quality (as measured by BLEU score) came from the application of a byte-pair encoding that reduced the vocabulary from 14,000 words to 2000 words.

The best neural model that they developed with that 100,000 word dataset scored 16.6 on German-to-English translation, while their phrase-based statistical model scored 15.9. For comparison, just two years earlier, with a 377,000 word English-to-Spanish dataset, Koehn and Knowles (2017) only obtained a BLEU score of 1.6 with a neural model, but 16.4 with a phrase-based statistical model.

Although the languages are different, the comparison seems valid because the better results required far less parallel text and because both pairs of researchers used recurrent neural networks. The difference was the algorithm that Sennrich et al. (2016) developed to replace the model's fixed vocabulary with a vocabulary of "subwords."

For example, the English present tense only has two forms – *speak* and *speaks* – while the Sicilian present tense has six – *parru*, *parri*, *parra*, *parramu*, *parrati* and *parranu*. But upon splitting them into subwords, *parr+* matches *speak+*, while the Sicilian verb endings (*+u*, *+i*, *+a*, *+amu*, *+ati* and *+anu*) match the English pronouns.

So in theory, subword splitting should allow us represent many different word forms with a much smaller vocabulary and should allow the translator to learn rare words and unknown words. For example, even if "jo manciu" ("I eat") does not appear at all in the dataset, but forms like "jo parru" ("I speak") and "iddu mancia" ("he eats") do appear, then subword splitting should allow the translator to learn "jo manciu" ("I eat").

In practice, achieving that effect required us to bias the learned subword vocabulary towards the desinences one

Table 1: Model Sizes

	defaults	our models	multilingual
layers	6	3	4
embedding size	512	256	384
model size	512	256	384
attention heads	8	4	6
feed forward	2048	1024	1536

finds in a textbook. Specifically, we added a unique list of words from the *Dieli Dictionary* and the inflections of verbs, nouns and adjectives from *Chiù dâ Palora* to the Sicilian data.

Because each word was only added once, none of them affected the distribution of whole words. But once the words were split, they greatly affected the distribution of subwords, filling it with stems and suffixes. So the subword vocabulary that the machine learns is similar to the theoretical stems and desinences of a textbook.

A Recipe for Low-Resource NMT

Even though we only have a little parallel text, we can still develop a reasonably good neural machine translator. We just have to to train a smaller model for the smaller dataset.

Training a large model on a small dataset is comparable to estimating a regression model with a large number of parameters on a dataset with few observations: It leaves you with too few degrees of freedom. The model thus becomes over-fit and does not make good predictions.

Reducing the vocabulary with subword-splitting, training a smaller network and setting a high-dropout parameter all reduce over-fitting. And self-attentional neural networks also reduce over-fitting because (compared to recurrent and convolutional networks) they are less complex. They directly model the relationships between words in a pair of sentences.

This combination of splitting, dropout and self-attention achieved a BLEU score of 25.1 on English-to-Sicilian translation and 29.1 on Sicilian-to-English with only 16,945 lines of parallel training data containing 266,514 Sicilian words and 269,153 English words.

And because the networks were small, each model took just under six hours to train on CPU.

Our success is an implementation of the best practices developed by Sennrich and Zhang (2019) with the self-attentional Transformer model developed by Vaswani et al. (2017). For training, we used the Sockeye toolkit by Hieber et al. (2017) running on a server with four 2.40 GHz virtual CPUs.

In their best practices for low-resource NMT, Sennrich and Zhang suggest the byte-pair encoding (i.e. subword-splitting) developed by (Sennrich et al., 2016), a smaller neural network with fewer layers, smaller batch sizes and larger dropout parameters.

As discussed above, a subword-splitting that reduced the vocabulary to 2000 words yielded their largest improvements in translation quality. But their most successful training also occurred when they set high dropout parameters.

During training, dropout randomly shuts off a percentage of units (by setting it to zero), which effectively prevents the units from adapting to each other. Each unit therefore becomes more independent of the others because the model is trained as if it had a smaller number of units, thus reducing over-fitting (Srivastava et al., 2014).

Subword-splitting and high dropout parameters helped us achieve better than expected results with a small dataset, but it was the Transformer model that pushed our BLEU scores into the double digits.

Compared to recurrent neural networks, the self-attention layers in the Transformer model more easily learn the

Table 2: Datasets and Results

dataset	subwords	lines	word count Sicilian	word count English	BLEU score En-Sc	BLEU score Sc-En
20	2,000	7,721	121,136	121,892	11.4	12.9
21	2,000	8,660	146.370	146,437	12.9	13.3
23	3,000	12,095	171,278	175,174	19.6	19.5
24	3,000	13,060	178,714	183,736	19.6	21.5
25	3,000	13,392	185,540	190,538	21.1	21.2
27	3,000	13,839	190,072	195,372	22.4	24.1
28	3,000	14,494	196,911	202,652	22.5	25.2
29	3,000	16,591	258,730	261,474	24.6	27.0
30	3,000	16,945	266,514	269,153	25.1	29.1
30	5,000	16,829	261,421	264,242	27.7	–
+back		+3,251	+92,141	–		
30	Sc: 5,000	16,891	262,582	266,740	19.7	26.2
Books	En: 7,500	32,804	–	929,043	35.1*	34.6*
+back	It: 5,000	+3,250	+92,146	–	* larger model	

dependencies between words in a sequence because the self-attention layers are less complex.

Recurrent networks read words sequentially and employ a gating mechanism to identify relationships between separated words in a sequence. By contrast, self-attention examines the links between all the words in the paired sequences and directly models those relationships. It's a simpler approach.

Combining these three features – subword-splitting, dropout and self-attention – yields a trained model that makes relatively good predictions. And as we assemble more parallel text, translation quality will improve even more.

Multilingual Translation

Our discussion so far has focused on a dataset of Sicilian-English parallel text. This section augments our dataset with parallel text in other languages to enable multilingual translation. It explains how we can train a single model to translate between multiple languages, including some for which there is little or no parallel text.

In our case, we can obtain Sicilian-English parallel text from the issues of *Arba Sicula*, but finding Sicilian-Italian parallel text is difficult. Nonetheless, our *Tradutturi Sicilianu* translates between Sicilian and Italian ("zero shot" translation) because we included Italian-English parallel text in our dataset.

To enable multilingual translation, we followed Johnson et al. (2016) and placed a directional token – for example, `<2it>` ("to Italian") – at the beginning of each source sequence. The directional token enables multilingual translation in an otherwise conventional model.

It's an example of transfer learning. In our case, as the model learns to translate from Italian to English, it would also learn to translate from Sicilian to English. And as the model learns to translate from English to Italian, it would also learn how to translate from English to Sicilian.

More parallel text is available for some languages than others however, so Johnson et al. also studied the effect on translation quality and found that oversampling low-resource language pairs improves their translation quality, but at expense of quality among high-resource pairs.

Importantly however, the comparison with bilingual translators holds constant the number of parameters in the model. Arivazhagan et al. (2019) show that training a larger model can improve translation quality across the board.

Our experience was consistent with their findings. As shown in Table 2, holding model size constant reduced translation quality when we added the Italian-English subset of Farkas' *Books* data (from the OPUS project) to our dataset.

So to push our BLEU scores into the thirties, we trained a larger model.

More broadly, Fan et al. (2020) developed the strategies to collect data for and to train a model that can directly translate between 100 languages. Previous efforts had resulted in poor translation quality in non-English directions because the data consisted entirely of translations to and from English.

To overcome the limitations of *English-centric* data, Fan et al. strategically selected pairs to mine data for, based on geography and linguistic similarity. Training a model on such a more multilingual dataset yielded very large improvements in translation quality in non-English directions, while matching translation quality in English directions.

Given such potential to expand the directions in which languages can be translated and to improve the quality with which they can be translated, an important question is what the model learns. Does it learn to represent similar sentences in similar ways regardless of language? Or does it represent similar languages in similar ways?

Johnson et al. examined two trained trilingual models. In one, they observed similiar representations of translated sentences, while in the second they noticed that the representations of zero-shot translations were very different.

Kudugunta et al. (2019) examined the question in a model trained on 103 languages and found that the representations depend on both the source and target languages and they found that the encoder learns a representation in which linguistically similar languages cluster together.

In other words, because similar languages learn similar representations, our model would learn Sicilian-English better from Italian-English data than from Polish-English data. And other Romance languages, like Spanish, would also be good languages to consider.

We can collect some of that parallel text from the resources at OPUS, an open repository of parallel corpora. Because it contains so many language resources, Zhang et al. (2020) recently used it to develop the OPUS-100 corpus, an *open-source* collection of English-centric parallel text for 100 languages.

Because it's a "rough and ready" massively multilingual dataset, it highlights some of the challenges facing massively multilingual translation. In particular, Zhang et al. show that a model trained with a vanilla setup exhibits *off-target translation* issues in zero-shot directions. In the English-centric case, that means the model often translates into the wrong language when not translating to or from English.

Zhang et al. tackle this challenge by simulating the missing translation directions. They first observe that Sennrich et al.'s (2015) method of back-translation "converts the zero-shot problem into a zero-resource problem" because it creates synthetic source language text. They then observe that this synthetic source language text simulates the missing translation directions.

The only obstacle is scalability. In a massively multilingual context, there are thousands of translation directions, which requires prohibitively many back-translations. To overcome this obstacle, Zhang et al. incorporate back-translation directly into the training process. And their final models exhibit improved translation quality and fewer off-target translation errors.

So we're excited about the potential for multilingual translation to improve translation quality and to create new translation directions for the Sicilian language.

Using the Italian-English subset of Farkas' *Books* data (from the OPUS project), we enabled zero-shot translation between Sicilian and Italian. And as shown in Table 2, multilingual translation greatly improve Sicilian-English translation quality, pushing our BLEU scores to 35.1 on English-to-Sicilian and to 34.6 on Sicilian-to-English.

Conclusion

Our recipe for low-resource neural machine translation – subword-splitting, dropout and self-attention – yields a trained model that makes relatively good predictions. And combining it with multilingual translation improves translation quality will improve even more.

So come to *Napizia* where we're developing Sicilian language resources and try our *Tradutturi Sicilianu*.

References

N. Arivazhagan, A. Bapna, O. Firat, D. Lepikhin, M. Johnson, M. Krikun, M. X. Chen, Y. Cao, G. F. Foster, C. Cherry, W. Macherey, Z. Chen, and Y. Wu. "Massively Multilingual Neural Machine Translation in the Wild: Findings and Challenges". *arXiv*, 2019. URL http://arxiv.org/abs/1907.05019.

J. K. Bonner. *Introduction to Sicilian Grammar*. Legas, Brooklyn, NY, 2001.

G. Cipolla. *Learn Sicilian, Mparamu lu sicilianu*. Legas, Mineola, NY, 2013.

A. Fan, S. Bhosale, H. Schwenk, Z. Ma, A. El-Kishky, S. Goyal, M. Baines, O. Celebi, G. Wenzek, V. Chaudhary, N. Goyal, T. Birch, V. Liptchinsky, S. Edunov, E. Grave, M. Auli, and J. Armand. "Beyond English-Centric Multilingual Machine translation". *arXiv*, 2020. URL https://arxiv.org/abs/2010.11125.

F. Hieber, T. Domhan, M. Denkowski, D. Vilar, A. Sokolov, A. Clifton, and M. Post. "Sockeye: A Toolkit for Neural Machine Translation. *arXiv*, 2017. URL http://arxiv.org/abs/1712.05690.

M. Johnson, M. Schuster, Q. V. Le, M. Krikun, Y. Wu, Z. Chen, N. Thorat, F. B. Viégas, M. Wattenberg, G. Corrado, M. Hughes, and J. Dean. "Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation". *arXiv*, 2016. URL http://arxiv.org/abs/1611.04558.

P. Koehn and R. Knowles. "Six Challenges for Neural Machine Translation". *arXiv*, 2017. URL http://arxiv.org/abs/1706.03872.

S. R. Kudugunta, A. Bapna, I. Caswell, N. Arivazhagan, and O. Firat. "Investigating Multilingual NMT Representations at Scale". *arXiv*, 2019. URL https://arxiv.org/abs/1909.02197.

R. Sennrich and B. Zhang. "Revisiting Low-Resource Neural Machine Translation: A Case Study". *arXiv*, 2019. URL http://arxiv.org/abs/1905.11901.

R. Sennrich, B. Haddow, and A. Birch. "Improving Neural Machine Translation Models with Monolingual Data". *arXiv*, 2015. URL http://arxiv.org/abs/1511.06709.

R. Sennrich, B. Haddow, and A. Birch. "Neural Machine Translation of Rare Words with Subword Units". *arXiv*, 2016. URL http://arxiv.org/abs/1508.07909.

N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. "Dropout: A Simple Way to Prevent Neural Networks from Overfitting". *Journal of Machine Learning Research*, 15(56):1929–1958, 2014. URL http://jmlr.org/papers/v15/srivastava14a.html.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. "Attention Is All You Need". *arXiv*, 2017. URL http://arxiv.org/abs/1706.03762.

B. Zhang, P. Williams, I. Titov, and R. Sennrich. "Improving Massively Multilingual Neural Machine Translation and Zero-Shot Translation". *arXiv*, 2020. URL https://arxiv.org/abs/2004.11867.