# Neural Machine Translation

## Do we need to start shivering in fear when we hear folks talking about neural machine translation?

There has been much in the news lately about the next wave of machine translation (MT) technology driven by something called deep neural nets (DNN). With the help of some folks who understand more about it than I do, I've attempted to provide a brief overview about what this is. First, I need to confess that I will be trying to explain something here that I don't fully understand myself. Still, I hope that my research has helped me comprehend and communicate some basic underlying principles.

Most lectures you've listened to about MT in the past few years have likely included a statement like this: "There are basically two kinds of MT—rule-based and statistical MT—and a third that combines the two—hybrid MT." You've also probably heard that rule-based MT was the earliest form of MT in the computer age, going back all the way to the 1950s. Back then, this form of MT consisted of a set of rules about the source and target language and included a dictionary. The transfer between the source and target language in rule-based MT happens either via an "interlingua," a computerized representation of the source text, or directly between the source and target language.

Statistical machine translation (SMT), on the other hand, became all the rage in the early 2000s. (The first commercial offering, LanguageWeaver [now owned by SDL], was launched in 2002; the widely used open-source engine Moses emerged in 2005; Google and Microsoft switched to statistical MT in 2007; and Yandex and Baidu started using SMT as recently as 2011.) SMT, or more accurately for all of these implementations, "phrase-based statistical machine translation," is trained on bilingual data and monolingual data. It parses the data into "n-grams," which are phrases consisting of an "n" number of words. The same thing happens to the source segment in the translation process. The source n-grams are then matched with

target n-grams, which are then combined to form whole segments again—and that's often where things go awry. (This is why SMT can prove to be a much richer resource when using an approach that just looks for fragments rather than whole segments.) Another potential flaw with SMT is the faulty selection process when the system tries to decide which of the many possible target n-grams to use. One way to guard against bad choices is by validating them on the basis of the monolingual target data on which the system was trained, but that only goes so far. (By the way, that's why an approach that offers access to more than just one of those n-gram fragments at a time within a translation environment tool has to be one of the up-and-coming developments.)

Neural machine translation (NMT)—and let's pause and be thankful that one of this technology's first proposed terms, "recursive hetero-associative memories for translation" (coined by Mikel L. Forcada and Ramon P. Neco in 1997) did not survive—is an extremely computing-power-heavy process (which is why it didn't go anywhere in 1997).[1] It's part of the larger field of "machine learning." In 1959, Arthur Samuel, a pioneer in the field of artificial intelligence and machine learning, defined machine learning as the "field of study that gives computers the ability to learn without being explicitly programmed."[2]

In SMT, the focus is on translated phrases that the computer is taught, which are then reused and fitted together according to statistics. NMT, on the other hand, uses neural networks that consist of many nodes (conceptually modeled after the human brain), which relate to each other and can hold single words, phrases, or any other segment. These nodes build relationships with each other based on bilingual texts with which you train the system. Because of these manifold and detailed relationships, it's possible to look at not just limited n-grams (as in SMT), but at whole segments or even beyond individual segments. This allows for the formation of significantly more educated guesses about the context, and therefore the meaning, of any word in a segment that needs to be translated. For instance, it's at least theoretically unlikely to have "Prince" translated as a (royal) prince by the NMT in a sentence like "The music world mourns the death of Prince," as Google, Microsoft, Yandex, and Baidu all do at the moment. (By the way, I'm mourning as well.)

This column has two goals: to inform the community about technological advances and at the same time encourage the use and appreciation of technology among translation professionals.

In languages like German with separable verbs, such as *umfahren* ("run over"), there is a much greater likelihood that the system will notice the missing part of the verb at the end of the sentence if the machine doesn't have to bother with chopping it into n-grams first. Take, for example, the simple sentence, *Ich fahre den Fußgänger um* ("I run over the pedestrian"). Bing translates it (today) as "I'm going to the pedestrian," and Google renders it as "I drive around the pedestrian." Only Yandex gets it right. (Baidu does not offer this language combination.)

Machine learning (itself a subfield of artificial intelligence) also comes into play as common usage gradually forges certain linguistic connections (e.g., "music world" and "Prince"; "fahren" and "um"). This means that, just as Arthur Samuel predicted, the computer continues to "learn" without explicitly being programmed.

At least theoretically, the NMT approach is very promising for generic engines like those of the search engines mentioned above (Google, Microsoft, Yandex, and Baidu). This is because "context" does not necessarily have to be specified by the training data, but can be recognized by the system evaluating the context (provided that the user supplies more than just a word or single phrase). So, you won't be surprised to hear that all those companies have already entered the realm of NMT. Naturally they don't reveal how much of their present system is "neural" versus "statistical only," but chances are it's a mix of both. And that would make all the more sense since one of the ways to use NMT is in combination with SMT—either as a quasi-independent verification process or as an integrated process that helps in selecting the "right" n-grams.

In some areas similar processes have already demonstrated remarkable success, including some that are used by search engines such as Google Image Search (which can be used very effectively for cross-language searches and be helpful in the translation process).

You probably read that Facebook launched its own MT system earlier this year specifically geared for the very casual language of its users. While that system is still mostly SMT-based, Facebook is working on an NMT solution as well. You might want to take a look at a presentation by Alan Packer, Facebook's director of engineering and language technology (formerly of Microsoft), entitled "Understanding the Language of Facebook."[3]

One misconception in Packer's presentation is his description of all this as a linear development. He paints SMT as more or less having run its course, now to be taken over by NMT. While I understand that someone so deeply embedded in one particular field must automatically think it the only worthwhile one, it's really unlikely to be the case. The same was said in the early days about rule-based MT (RbMT) by proponents of SMT, and that assumption has not proven to be true. Many systems are using a hybrid approach between SMT and RbMT, and for some language combinations RbMT might still be a better solution (especially for language pairs that are very close to each other, like Catalan and Spanish or Croatian and Serbian).

But are we on the verge of a big new breakthrough overall? To answer that, you might want to look through the joint presentation by Diego Bartolome (Tauyou Language Technology) and Gema Ramirez (Prompsit Language Engineering), "Beyond the Hype of Neural Machine Translation."[4] Since there is no open-source toolkit for NMT, like Moses for SMT, very few companies actually offer customized NMT systems. There are components like the deep learning frameworks Theano and Torch and specific NMT software like GroundHog and seq2seq, but these are anything but user-friendly and require significant expertise. Using them to build the NMT engine takes a lot of computing power (10 CPUs or 1 GPU—graphics processing unit) and time (about two weeks of training per language pair once the training data is assembled and cleaned). Tauyou Language Technology and Prompsit Language Engineering are some of the first vendors who are working on commercial versions of NMT. (Interestingly, Tauyou comes with an SMT background, and Prompsit with a background in RbMT). While they are not actively selling the NMT solutions yet, they are doing a lot of pilots, as you'll see from the presentation. The results of these pilots are mixed.

I already mentioned the much larger processing and time requirements. There are also limitations as far as the number of words per language that can be trained with the processing power currently available to mere mortals (in opposition to companies like Google), the approximately three-fold time the system takes to actually translate, and the fact that retraining the system with new data would once again take two weeks. But there are some improvements in the quality—although, according to the presentation, this is not adequately appreciated by translators. (I assume this has to do with even less predictability when it comes to post-editing—and presumably even more erratic decisions when it comes to partial suggestions.) However, this is still very early in the game, so I wouldn't be surprised to see the quality continue to improve.

So, do we need to start shivering in fear when we hear folks talking about NMT? Although I don't completely understand the technology, I (and now you) have seen numbers showing only moderate progress. So, no, we'll continue to be very assured of our jobs for a long time. I do look forward, though, to seeing how NMT will creatively find its way into our translation environment and improve our work.

**NOTES**


1. Forcada, Mike L., and Ramon P. Neco. "Recursive Hetero-Associative Memories for Translation," In *Biological and Artificial Computation: From Neuroscience to Technology* (Springer, 1997), 453–462, http://bit.ly/2b1NSOp.
2. Samuel, Arthur, "Some Studies in Machine Learning Using the Game of Checkers," *IBM Journal* (July 1959), http://bit.ly/2aX2sF9.
3. Packer, Alan. "Understanding the Language of Facebook," bit.ly/2aeIzt2.
4. Bartolome, Diego, and Gema Ramirez. "Beyond the Hype of Neural Machine Translation," *MIT Technology Review* (May 23, 2016), bit.ly/2aG4bvR.


**Jost Zetzsche** is the co-author of *Found in Translation: How Language Shapes Our Lives and Transforms the World*, a robust source for replenishing your arsenal of information about how human translation and machine translation each play an important part in the broader world of translation. Contact: jzetzsche@internationalwriters.com.