

## Google's SyntaxNet: Achieving Better Syntax Recognition Using Neural Networks

Are computers smarter than us? It is commonly acknowledged that the rise of personal computing has outclassed humans in many areas: chess, computation, counting, data recall and many more. However, there is one area where humans maintain an edge, and that is in the understanding and use of organic language. While great strides have been made in the pursuit of natural language supremacy, few technologies have been able to accurately capture the nuance of human speech. With Google's state-of-the-art natural language processing architecture, SyntaxNet, the company seeks to draw closer to machine mastery of human language.

Unveiled in May of 2016, Google's SyntaxNet framework and its correspondent parser elegantly known as Parsey McParseface, is among the most sophisticated English natural language processing technologies available today. The framework, which is open source and makes use of neural network technologies (*Petrov 2016*) includes code to train the model on a particular dataset as well as a more general-use case parser by the name of ParseyMcParseface. Claimed to be the most precise natural language processing model in the world according to Stanford University's Daniel Andor (*Globally Normalized Transition-Based Neural Networks 2016*), the model emphasizes the importance of global normalization rather than local to avoid label bias. But first of all, what is a neural network, and why would it be useful for natural language processing?

The term 'neural network' was originally coined by a pair of University of Chicago professors back in the 1940s who went on to found the first cognitive science department at MIT in 1952 (*Hardesty 2017*). Largely forgotten until the turn of the century, the surge of cheaper and more powerful graphical processing units led to a renewed interest in the idea of modelling data problems using machine learning to 'train' algorithms using carefully curated data in a similar fashion as the human brain (*Hardesty 2017*). Using a vast multitude of computing cores, data flows through each in a single direction in order of least to most complex. Then, using a series of weights and threshold values – usually set at random to start – training data is processed and fed through successive layers of computation until data that has similar labels is grouped together. The areas of the network that lead to successful labeling are strengthened and those that do not are weakened in terms of emphasis, similar to how the human brain learns new information and behaviors.

Initially conceived as an area of neuroscience, the neural nets as implemented today are proof of the fact that the human brain is in fact a computational machine in addition to powering our consciousness and many of the other critical facets of the human experience. Thanks in part to the PC gaming industry, whose graphical demands have generated graphics demands that contain hundreds or even thousands of processing units linked together, the opportunity to test out real-life use cases for these networks grew exponentially. Where there were once primitive single-digit layer networks before the turn of the century, state-of-the-art GPUs have the capacity to generate up to 50 layer networks that will occasionally generate stunning results, such as one of Google's first forays into neural networks: DeepDream.

Released to the public in 2015, the software was originally intended to detect objects and other patterns in images. By running it in reverse, you can create a surrealistic interpolation of an image by

specifying what you are looking for (*Chandler 2015*). Such an algorithm is analogous to looking for animal shapes or other constellations in the night sky. At its most extreme, the process can be coupled with virtual reality to simulate hallucinatory experiences (*Suzuki 2017*). That said, the training set was conducted on a subset of labeled dog breeds so it had limited practical use for Google's primary business activities

In a nod to Google's core business, neural networks were next applied to Google's in-house natural language processing engine. Given that the vast majority of the company's revenue comes from search -- and by extension advertisement -- revenue, improving the ability to understand grammatical meaning was a natural fit for this technology. It works by taking a sentence as an input, tagging each word as a part of speech like most taggers, but with the additional layer of dependency trees that enable encoding of modifiers like adverbs and prepositional phrases that enable more sophisticated understanding. While contemporary parsers are already quite adept at identifying these parts of speech, its implementation of dependency trees is part of what makes SyntaxNet unique. The ambiguities inherent in English make it such that prepositions are not necessarily adjacent to the words they modify. This lack of clarity can make it difficult for computers to discern meaning from phrases that seem absurd to native English speakers. For example, "John skied down the mountain in his snowmobile" has intuitive meaning to most English-speakers, but a machine runs the risk of absurdly presuming that there is a mountain in John's snowmobile. The problem intensifies with longer sentences due to the increased number of possible combinations, and the parser may struggle to get rid of these in a timely manner. Depending on the complexity of the sentence, a single sentence of two dozen words could have as many different syntactic interpretations for a given context (*Houpt 2006*).

SyntaxNet uses neural networks in these instances to assign plausibility scores for each combination from left to right. It maintains multiple hypotheses as long as there are no higher-scored hypotheses to compete with it (*Petrov 2016*). By integrating this with its learning algorithm, the end result is a sophisticated decision-making algorithm that works very similarly to the human brain's decision-making process, yielding up to 94% accuracy (*Bohn 2016*). Given that professional linguists achieve consensus in roughly 96% of examples, this performance is an impressive feat.

Still, the technology is far from the final word on syntax parsing. The technology still fails in situations with ambiguous prepositional phrases that necessitate real, functional knowledge of our reality. This also does not capture additional ambiguities present in slang and other forms of informal speech, like often presented in social media. In fact, sentences drawn from the web at random only achieve about 90% accuracy according to Google's own statistics (*Petrov 2016*). Only providing a built-in parser for the English language is also a challenge, given that roughly 40% of the web is written in another language (*Usage statistics of content languages for websites 2020*).

That said, the parsing engine is still useful for contextual interpretations and is continuously being improved. In 2017, Google developed a major upgrade that allows dynamic creation of neural networks in real-time during the processing of a phrase as well as a new parser called ParseySaurus, which includes support for additional languages beyond English (*Weiss 2017*). This technology relies on character-based input to account for differences in spelling and context to achieve even higher accuracy and better speed than its contemporaries. On the other hand, alternatives have sprung up since its release such as StanfordNLP/Stanza -- which also makes use of neural networks -- and spaCy, which has high utilization rates in industry due to much lower processing power requirements. These tools offer more flexible utility than Google's offering as well as improved tokenization interfaces. All of these tools have powerful implications for applications to enable greater efficiencies for private industry and end-users alike. Until

this lofty goal is complete, however, ambitious researchers and developers will likely continue to make efforts to bring natural language to the machine world.

## Bibliography

Andor, D., Alberti, C., Weiss, D., Severyn, A., Presta, A., Ganchev, K., . . . Collins, M. (2016, June 08). Globally Normalized Transition-Based Neural Networks. Retrieved November 05, 2020, from <https://arxiv.org/abs/1603.06042v2>

Bohn, D. (2016, May 12). Google is giving away the tool it uses to understand language, Parsey McParseface. Retrieved November 05, 2020, from <https://www.theverge.com/2016/5/12/11666414/google-parsey-mcparseface-tensorflow-open-source-language-tool>

Chandler, N. (2015, September 01). How Google Deep Dream Works. Retrieved November 05, 2020, from <https://computer.howstuffworks.com/google-deep-dream.htm>

Hardesty, L. (2017, April 14). Explained: Neural networks. Retrieved from <https://news.mit.edu/2017/explained-neural-networks-deep-learning-0414>

Petrov, S. (2016, May 12). Announcing SyntaxNet: The World's Most Accurate Parser Goes Open Source. Retrieved November 05, 2020, from <https://ai.googleblog.com/2016/05/announcing-syntaxnet-worlds-most.html>

Sutton, J. (2020, June 23). Advanced grammar and Natural Language Processing with Syntaxnet. Retrieved November 05, 2020, from <https://algorithmia.com/blog/advanced-grammar-and-natural-language-processing-with-syntaxnet>

Suzuki, K., Roseboom, W., Schwartzman, D.J. et al. A Deep-Dream Virtual Reality Platform for Studying Altered Perceptual Phenomenology. *Sci Rep* 7, 15982 (2017). <https://doi.org/10.1038/s41598-017-16316-2>

Usage statistics of content languages for websites. (2020, November). Retrieved November 05, 2020, from [https://w3techs.com/technologies/overview/content\\_language](https://w3techs.com/technologies/overview/content_language)

Weiss, D., & Petrov, S. (2017, March 15). An Upgrade to SyntaxNet, New Models and a Parsing Competition. Retrieved November 05, 2020, from <https://ai.googleblog.com/2017/03/an-upgrade-to-syntaxnet-new-models-and.html>