Rijul Tikku

Statistical Language Models Tech Review

Just a few short years ago, in early iterations of voice assistants, one had to ask specifically worded queries over and over until it finally understood what was said. Now, not only can a phone recognize when someone is speaking to it and answer your question completely hands-free, but one can talk to it like it's another human, not a computer. This is because of the advancements made in Statistical Language Models(LMs) and natural language processing in recent years. To understand how LMs have been successful in changing human-technology interaction this paper will discuss a brief overview of Statistical Language Models, go in-depth into Neural Language Models (NLMs), and explore how Neural Language Models have recently advanced and where they are moving.

The basis of Statistical Language Models is developing probabilistic models to predict words, phrases, and sentences using context. There are many ways that the probabilistic models are made based on training data. Some notable techniques include N-gram, Exponential, and Neural Models. N-Gram models train from data using n-grams which is a sequence of some variable n items that are used to train the model. Exponential models use n-grams along with other parameters that try to find the distribution with the most entropy. These models have some flaws including sparse training data due to the nature of n-grams, the exponential growth of vocabulary, and sensitivity to training data. Neural modeling is a newer technique that addresses some of the shortcomings of the previous types of models and has been shown to be more effective in a variety of applications.

Neural Language Models are models based on neural networks to create a network of nodes and weighted edges to compute the probability of the next item in the sequence. An important feature of the nodes is that they are grouped by a variety of features created by the network which may be along the lines of connotation, denotation, gender, plurality, and more. Instead of providing weights to just individual terms, terms are grouped. This creates a distributed representation rather than a local one. Having a distributed representation solves many issues that were previously seen: overfitting to training data is less of a problem when instead of computing with specific words there are a set of features that a word may fall into; since the neural model is almost a function due to the nature of neural networks, we do not need to store all of the n-gram counts that grow exponentially but only the set of features and how

they interact; sparse training data is also less of an issue with neural networks. The main advantage comes from its ability to generalize, and when dealing with natural language processing, this is an important task as the sentences can be very diverse. This method does have its weaknesses. The computation times can be expensive, the neural networks used are too shallow, and using long-term context can be unreliable. However, these issues have been constantly worked on; Even within neural networks, there is a lot of variety in implementation which has been explored more and more as time has gone on.

Recently, Neural language models have been shown to outperform the classical methods like the ones mentioned previously. One notable variation of neural language modeling comes from the use of Recurrent Neural Networks or RNNs. An RNN is a neural network that is both fully connected and loops to iteratively add or change the same weights when needed to be dynamic. The benefit for RNNs is that they can deal with variable length inputs which makes them especially useful for natural language LMs. Because the weights can be looped, the network can predict a word at the end of any various-length sequence much more efficiently. RNNs can have Long Short Term Memory which means its short-term memory can be adapted over its iterations to reflect longer sequences. This method fixes the issues of long-term context and computational complexity. Drawbacks to this method include that longer sequences cannot be parallelized so training times can be very high because it is sequential. Another recent technique that addressed the flaws was developed by Google called BERT. BERT uses a transformer model which is similar to RNNs but can process all of the input at one time and can take advantage of parallelization. This all comes from the attention mechanisms of transformer models which are layers that allow one to get the weights from any of the previous states which dramatically increases performance.

There are many more players in the race for the most advanced language model including GPT-3 by OpenAI. There are still things to improve and as more people continue to innovate, the technology will only get better. Many computer science journalists have said language processing will likely be the first major step to true Artificial Intelligence considering where we are at now. The implications of this technology improving can benefit people around the world whether it autocompleting tasks, mining data, or even creating art.

References

(1) https://www.cs.cmu.edu/~roni/papers/survey-slm-IEEE-PROC-0004.pdf

(2) https://arxiv.org/pdf/2111.01243.pdf

(3) http://www.scholarpedia.org/article/Neural_net_language_models#:~:text=A%20neural%20network%20language%20model,of%20the%20curse%20of%20dimensionality.

(4) https://towardsdatascience.com/neural-language-models-32bec14d01dc