news

Science | DOI:10.1145/2874915

Gregory Goth

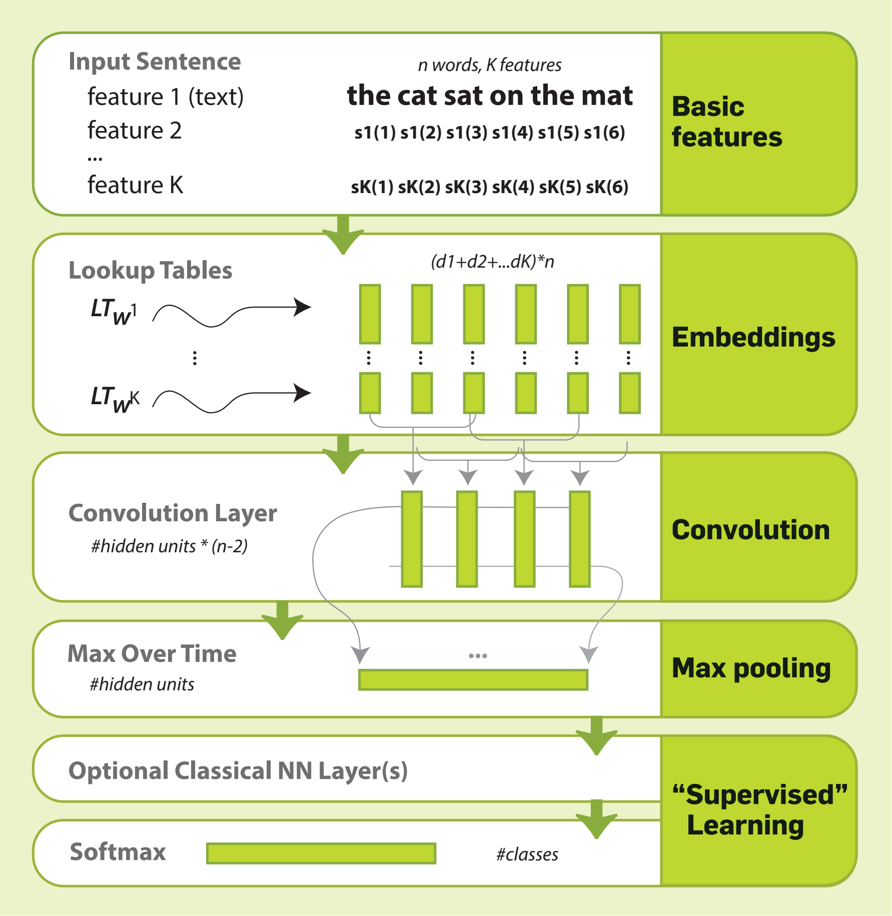
# Deep or Shallow, NLP Is Breaking Out

*Neural net advances improve computers' language ability in many fields.*

One of the featured speakers at the inaugural Text By The Bay conference, held in San Francisco in April 2015, drew laughter when describing a neural network question-answering model that could beat human players in a trivia game.

While such performance by computers is fairly well known to the general public, thanks to IBM's Watson cognitive computer, the speaker, natural language processing (NLP) researcher Richard Socher, said, the neural network model he described "was built by one grad student using deep learning" rather than by a large team with the resources of a global corporation behind them.

Socher, now CEO of machine learning developer MetaMind, did not intend his remarks to be construed as a comparison of Watson to the academic model he and his colleagues built. As an illustration of the new technical and cultural landscape around NLP, however, the laughter Socher's comment drew was an acknowledgment that basic and applied research in language processing is no longer the exclusive province of those with either deep pockets or strictly academic intentions.



**General Deep Architecture for NLP. Source: Collobert & Weston, Deep Learning for Natural Language Processing, 2009 Conference on Neural Information Processing Systems.**

Indeed, new tools and new techniques—particularly open source technologies such as Google's word2vec neural text processing tool—combined with steady increases in computing power, have broadened the potential for natural language processing far beyond the research lab or supercomputer. In domains as varied as finding pertinent news for a company's potential investors to making hyper-personalized recommendations for online shopping to making music recommendations on streaming radio services, NLP is enabling everyday human-computer interaction in an ever-increasing range of venues. In the process, some of these advances are not only redefining what computers and humans can accomplish together, but also the very concept of what deep learning is.

### Vectors Deep or Wide

One approach to natural language processing that has gained enormous traction in the past several years is representing words as vectors—that is, each word is given a series of scores that position it in an arbitrary space. This principle was explained by deep learning pioneer Geoffrey Hinton at a recent presentation to the Royal Society, the U.K. national academy of science. Hinton, a distinguished researcher for Google and distinguished professor emeritus at the University of Toronto, said, "The first thing you do with a word symbol is you convert it to a word vector. And you learn to do that, you learn for each word how to turn a symbol into a vector, say, 300 components, and after you've done learning, you'll discover the vector for Tuesday is very similar to the vector for Wednesday."

The result, Hinton said, is that given enough data, a language model can then generalize: for any plausible sentence with Tuesday in it, there's a similar plausible sentence with Wednesday in it. More broadly, words with similar vector scores can be used to classify and cluster concepts. Companies using vector-based NLP technologies in production analyze concepts as varied as documents referring to a business's financial activity or fashion customers' reviews of a piece of clothing to try to help predict what type of customer will gravitate toward a certain style, much more quickly than could active human curation alone.

**NLP is enabling everyday human-computer interaction in an ever-increasing range of venues.**

In a recent interview with *Communications*, Hinton said his own research on word vectors goes back to the mid-1980s, when he, David Rumelhart, and Ronald Williams published work in *Nature* that demonstrated family relationships as vectors. "The vectors were only six components long because computers were very small then, but it took a long time for it to catch on," Hinton said.

The concept has indeed caught on, and as explained by the Google team that recently released the open source TensorFlow machine learning system, using the vector principle in NLP helps to address problems caused by methods that treat words as discrete atomic symbols. That discrete classification leads to data sparsity, and usually means more data may be needed to successfully train statistical models. Using vector representations can overcome some of these obstacles.

In introducing their explanation, the TensorFlow team cited image processing as a field that already used vectors of raw pixel intensities: that field is also one of the foremost examples of using "deep" neural networks, networks of multiple layers that learn from each other as data is passed between them, to improve accuracy and performance.

As vectors became more popular in NLP research, so too did the principles of deep learning within the field. However, orthodox deep learning approaches that may be very suitable for raw pixel intensities can prove problematic for text; as explained by data scientist Will Stanton in a presentation prepared for the 2015 machine learning "Ski Hackathon," each hidden layer and each feature means more parameters to train, and human-generated text has a near-infinite number of features and data.

In 2013, however, a research team from Google led by Tomas Mikolov published word2vec, a three-layer model (input, hidden layer, and output layer) that vastly improved the speed of what had been the contemporary state of the art—by making the neural network shallower but wider.

"Shallow models can be trained using a bigger, wider, net on much more data, which can pay off in some cases much more than training a deeper net on a small subset of the training data, due to time constraints—the training can be very expensive," Mikolov, now a research scientist at Facebook, said. For example, he said, one of the first well-known examples of a vectorized neural network contained 50 dimensions; that is, just 50 neurons were used.

"It took two months to train this model on approximately 600 million words," he said. "In my papers, I analyzed the shallow net's performance and on some tasks, going to 200–300 dimensionality helps—that is, the model is wider, and more precise; also, using the shallow model and an efficient implementation, I could train the word vectors with word2vec on a 100-billion-word dataset in hours.

"If you pre-train the vectors—convert words into distributed continuous vectors that capture in some sense the semantics of the words—on Wikipedia, that is several billions of words you just trained the model on. The resulting vectors are not good by themselves for anything concrete. Then, when you pick a task of, say, sentiment analysis, you can build a classifier that will take these pre-trained vectors as its input, instead of just the raw words, and perform classification. This is because labeled examples are much more expensive to obtain than the unlabeled ones. The resulting classifier can work much better when it is based on the pre-trained word features."

Word2vec relies on two algorithms, one a "continuous bag of words," a model trained to predict a missing word in a sentence based on the surrounding context; the other deemed "skip-gram," which uses each current word as an input to a log-linear classifier to predict words within a certain range before and after that current word.

While Mikolov's flattening of the neural network concept appears on the surface to be a significant break from other approaches to NLP, Yoav Goldberg and Omer Levy, researchers at Bar-Ilan University in Ramat-Gan, Israel, have concluded much of the technique's power comes from tuning algorithmic elements such as dynamically sized context windows. Goldberg and Levy call those elements hyperparameters.

"The scientific community was comparing two implementations of the same idea, with one implementation —word2vec—consistently outperforming another, the 'traditional' distributional methods from the 90's," Levy said. "However, the community did not realize that these two implementations were in fact related, and attributed the difference in performance to something inherent in the algorithm.

"We showed that these two implementations are mathematically related, and that the main difference between them was this collection of 'hyperparameters'. Our controlled experiments showed that these hyperparameters are the main cause of improved performance, and not the count/predict nature of the different implementations."

Other researchers have released vectorization technologies with similar aims to word2vec's. For example, in 2014, Socher, then at Stanford University, and colleagues Jeffrey Pennington and Christopher D. Manning released Global Vectors for Word Representation (GloVe). The difference between GloVe and word2vec was summarized by Radim Rehurek, director of machine learning consultancy RaRe technologies, in a recent blog post:

"Basically, where GloVe precomputes the large word x word co-occurrence matrix in memory and then quickly factorizes it, word2vec sweeps through the sentences in an online fashion, handling each co-occurrence separately," Rehurek, who created the open source modeling toolkit gensim and optimized it for word2vec, wrote. "So, there is a trade-off between taking more memory (GloVe) vs. taking longer to train (word2vec)."

Machine learning specialists in industry have already taken to using general-purpose tools such as GloVe and word2Vec, but are not getting caught up in comparisons.

"There have definitely been some arguments about the kinds of results that have been presented, like the accuracy of GloVe vs. word2vec," said Samiur Rahman, senior machine learning engineer at MatterMark, a company specializing in document search for business news. "And then Levy and Goldberg have their own vectors too, but they're all essentially pretty good for general-purpose vectors. So instead of spending time trying to figure out which of the three works best for you, I would recommend—and it's worked well for us—choose the one that has the best production implementation right now, that fits easily into your workflow, and also figure out which one has better tools to train on data you have."

Rahman and others maintain word2vec, while very useful in initializing domain-specific NLP, complements but does not supplant other models. "We used word2vec to construct document vectors, because we didn't have a lot of labeled examples of what were funding articles and what weren't," Rahman said, adding that once a given model within a narrow domain has enough training data, a Naive Bayes-based model works well, with less computational complexity.

## What's + Next + NLP = ?

Just as "Hello, World" may be the best-known general programming introductory example, Mikolov, who was then at Microsoft Research, also introduced what fast became a benchmark equation in natural language processing at the 2013 proceedings of the North American Association for Computational Linguistics, the *king-man+woman=queen* analogy, in which the computer solved the equation spontaneously.

"What was really fascinating about it was that nobody trained the computer to solve these analogies," Levy said. "It was a by-product of an unsupervised learning scheme. Word2vec shows it, but also a previous model of Mikolov's shows it as well. So you would train the computer to do language modeling, for example, or to complete the sentence and you would get vectors that exhibit word similarity like 'debate and discussion,' or 'dog and cat,' but nobody told

# ACM Member News

### RESOLVING MATH, CS ISSUES WITH DISTRIBUTED COMPUTING

**Mathematics and problem-solving are two of Faith Ellen's lifelong passions.**

**A professor of computer science (CS) at the University of Toronto, Ellen has ample opportunity to use mathematical and problem-solving skills in her research involving distributed data structures and the theory of distributed computing. "I've always liked the fact that there were right and wrong answers to questions and problems," she says.**

**Ellen recalls an early affinity for CS, taking courses in theory and learning to program in high school. She received her bachelor's degree with honors in mathematics and computer science at the University of Waterloo, where she also received her master's degree in CS on formal CS programming languages; "I thought they were beautiful." She earned her doctorate from the University of California, Berkeley with a dissertation on "lower bounds for cycle detection and parallel prefix sums."**

**Her current research involves distributed computing and proving lower bounds on "the complexity of concrete problems to understand how the parameters of various models affect their computational power."**

**Among her proudest accomplishments was co-authoring "Impossibility Results for Distributed Computing" with Hagit Attiya, which was published as a book in 2014. The pair surveyed results from various distributed computing models proving tasks to be impossible, either outright or within given resource bounds. "Lower bounds are fun. I like to prove that it's impossible to do something faster or that it's impossible to do something without using lots of storage space."**

***—Laura DiDio***

it anything about analogies, and the fact these analogies emerged spontaneously was amazing. I think it's the only case where we use 'magic' in a science publication because it looked like magic."

It was not magic, of course, but the principle behind it allows the concept of vectorization to be made very clear to those far outside the NLP and machine learning communities. As data scientist Chris Moody explained, also at the 2015 Text By The Bay conference, the gender-indicating vectors for king and queen will be the same length and angle as those for woman and man, aunt and uncle, and daughter and son; in fact, any conceptual group at all, such as different languages' words for the same animal, or the relationship of countries and their capital cities, can be shown to have similar properties that can be represented by similar vectors—a very understandable universality.

"That's the most exciting thing, lighting up that spark," he told *Communications*. "When people say, 'oh, you mean computers understand text? Even at a rudimentary level? What can we do with that?' And then I think follows an explosion of ideas."

Moody works for online fashion merchant Stitch Fix, which uses analysis of detailed customer feedback in tandem with human stylists' judgments to supply its clients with highly personalized apparel. The Stitch Fix experience, Moody said, is not like typical online shopping.

"Amazon sells about 30% of their things through personalized recommendations—'People like you bought this'—and Netflix sells or rents out 70% of their viewings through those kinds of recommendations. But we sell everything through this personalized service. The website is very minimal. There's no searching, no inventory, no way for you to say 'I want to buy item 32.' There is no fallback; we have to get this right. So for us, being on the leading edge of NLP is a critical differentiating factor."

The combination of the company's catalog and user feedback—for example, a certain garment's catalog number and the word "pregnant" and words that also denote pregnancy or some sort of early-child-rearing status, located near each other in the Stitch Fix algorithm's vector space—can help guide a stylist to supply a customer with a garment similar in style to the original, but cut for maternity wear. What is more, he said, the word2vec algorithm as used by Stitch Fix in production is not used on text.

"Most of our reasoning is by analogy; it's not logical reasoning."

"In our boxes we ship five items. and you can use word2vec here and say 'given this item in that box, can you predict the other items?'" Moody said. "Word2vec doesn't care if it's a word or not a word, it's just another token and you have to make that token similar to the other tokens. It's almost like using the backbone of the word2vec algorithm to look inside someone's closet and saying 'these things are very similar because they would all appear in the same sort of closet together.'"

In fact, he said, the company is starting to use analogical principles to go beyond text and synthesize the images of imagined new items from images of existing pieces of clothing—a process he said was "starting to get toward this hint of creativity. So if you think of these word vectors like *king-man+woman=queen*, we're now exploring spaces between those data points, and that's what we're calling creativity—things that have never been seen before, but are really just somewhere in between all those other observations."

How quickly that sort of creativity may lead to breakthroughs for machine learning and artificial intelligence is clearly an open question, but it bears mulling, given an observation about the basis of human reasoning from Hinton.

"Most of our reasoning is by analogy; it's not logical reasoning," he said. "The early AI guys thought we had to use logic as a model and so they couldn't cope with reasoning by analogy. The honest ones, like Allen Newell, realized that reasoning by analogy was a huge problem for them, but they weren't willing to say that reasoning by analogy is the core kind of reasoning we do, and logic is just a sort of superficial thing on top of it that happens much later." ■

### Further Reading

*Levy, O. and Goldberg, Y.,*
**Linguistic Regularities in Sparse and Explicit Word Representations.** ***Proceedings of the 18th Conference on Computational Natural Language Learning, 2014.***
**http://bit.ly/1OXBici**

*Mikolov, T., Chen, K., Corrado, G., and Dean, J.,*
**Efficient Estimation of Word Representations in Vector Space.** ***Proceedings of Workshop at International Conference on Learning Representations, 2013,***
**http://arxiv.org/abs/1301.3781**

*Goldberg, Y., and Levy, O.,*
**word2vec Explained: Deriving Mikolov et al.'s Negative-Sampling Word-Embedding Method,** ***arXiv 2014.*** **http://arxiv.org/abs/1402.3722**

*Pennington, J., Socher, R., and Manning, C.,*
**GloVe: Global Vectors for Word Representation.** ***Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing.***
**http://nlp.stanford.edu/projects/glove/**

*Moody, C.*
**A Word is Worth a Thousand Vectors.** ***MultiThreaded, StitchFix, 11 March 2015.***
**http://bit.ly/1NL35xz**

*Iyyer, M., Boyd-Graber, J., Claudino, L., Socher, R., and Daume III, H.,*
**A Neural Network For Factoid Question Answering Over Paragraphs.** ***Proceedings of EMNLP 2014***
**https://cs.umd.edu/~miyyer/qblearn/**

### Video resources

*Hinton, G.*
**Deep Learning.** ***Royal Society keynote, recorded 22 May 2015.***
**https://www.youtube.com/watch?v=IcOMKXAw5VA**

*Socher, R.*
**Deep Learning for Natural Language Processing.** ***Text By The Bay 2015.***
**https://www.youtube.com/watch?v=tdLmf8t4oqM**

**Bob Dylan and IBM Watson on Language,** ***advertisement, 5 October 2015.***
**https://www.youtube.com/watch?v=pwh1INne97Q**

**Gregory Goth** is an Oakville, CT-based writer who specializes in science and technology.