



# NLP 2019 Highlights

by Elvis Saravia

Email [ellfae@gmail.com](mailto:ellfae@gmail.com)

Twitter: [omarsar0](https://twitter.com/omarsar0)

LinkedIn: <https://www.linkedin.com/in/omarsar/>

Medium: <https://medium.com/@ibelmopan>

2019 was an impressive year for the field of natural language processing (NLP). In this report, I want to highlight some of the most important stories related to machine learning and NLP that I came across in 2019. I will mostly focus on NLP but I will also highlight a few interesting stories related to AI in general. The headlines are in no particular order. Stories may include publications, engineering efforts, yearly reports, the release of educational resources, etc.

## Table of Contents

● Publications -----	3
● ML/NLP Creativity and Society -----	7
● ML/NLP Tools and Datasets -----	12
● Articles and Blog Posts -----	16
● Ethics in AI -----	21
● ML/NLP Education -----	22

## Publications

Google AI introduces [ALBERT](#) which is a lite version of [BERT](#) for self-supervised learning of contextualized language representations. The main improvements are reducing redundancy and allocating the model's capacity more efficiently. The method advances state-of-the-art performance on 12 NLP tasks.

Earlier this year, researchers at NVIDIA published a [popular paper](#) (coined StyleGAN) which proposed an alternative generator architecture for GANs, adopted from [style transfer](#). Here is a [follow-up work](#) where that focuses on improvements such as redesigning the generator normalization process.



Our configuration F — generated images



Our configuration F — real images

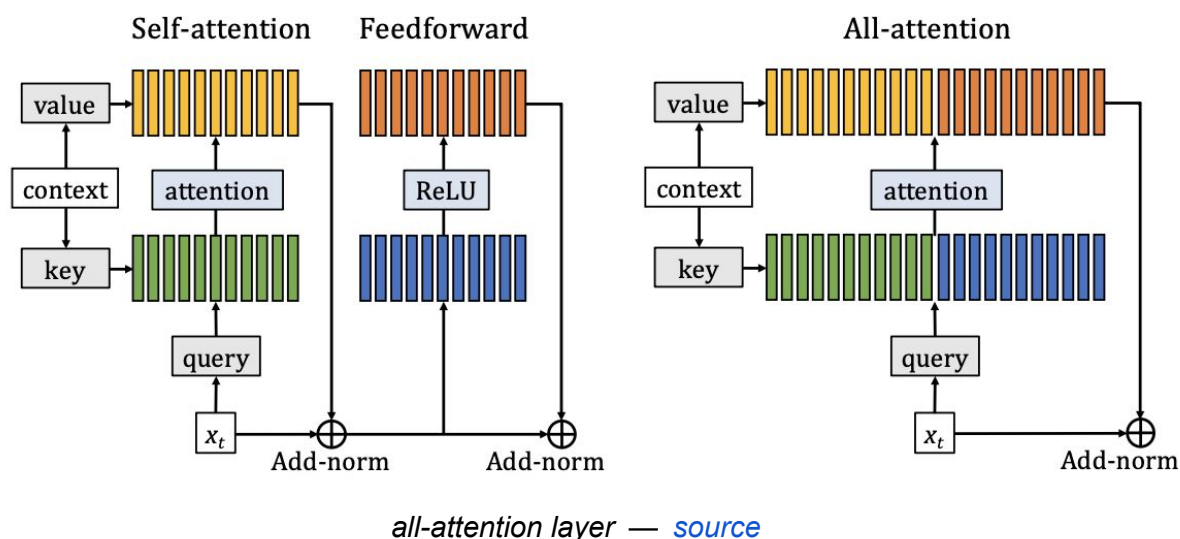
*the top row shows target images and the bottom row shows synthesized images — [source](#)*

One of my favorite papers this year was [code2seq](#) which is a method for generating natural language sequences from the structured representation of code. Such research can give way to applications such as automated code summarization and documentation.

Ever wondered if it's possible to train a biomedical language model for biomedical text mining? The answer is [BioBERT](#) which is a contextualized approach for extracting important information from biomedical literature.

After the release of BERT, Facebook researchers published [RoBERTa](#) which introduced new methods for optimization to improve upon BERT and produced state-of-the-art results on a wide variety of NLP benchmarks.

Researchers from Facebook AI also recently published a [method](#) based on an all-attention layer for improving the efficiency of a Transformer language model. More work from this research group includes a [method](#) to teach AI systems on how to plan using natural language.



Explainability continues to be an important topic in machine learning and NLP. This [paper](#) provides a comprehensive overview of works addressing explainability, taxonomies, and opportunities for future research.

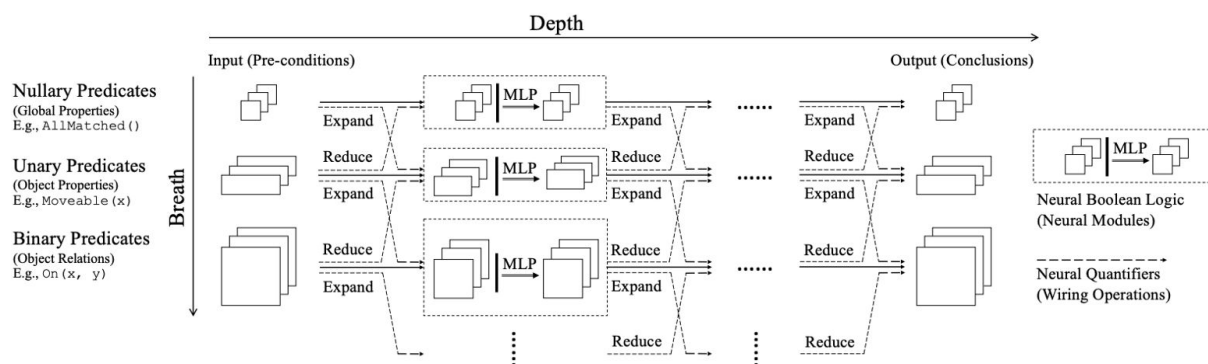
Sebastian Ruder published his [thesis](#) on Neural Transfer Learning for Natural Language Processing.

A group of researchers developed a [method](#) to perform emotion recognition in the context of conversation which could pave the way to affective dialogue generation. Another related work involves a GNN approach called [DialogueGCN](#) to detect emotions in conversations. This research paper also provides [code implementation](#).

The Google AI Quantum team published a [paper](#) in Nature where they claim to have developed a quantum computer that is faster than the world's largest supercomputer. Read more about their experiments [here](#).

As mentioned earlier, one of the areas of neural network architectures that require a lot of improvement is explainability. This [paper](#) discusses the limitations of attention as a reliable approach for explainability in the context of language modeling.

[Neural Logic Machine](#) is a neural-symbolic network architecture that is able to do well at both inductive learning and logic reasoning. The model does significantly well on tasks such as sorting arrays and finding shortest paths.



Neural logic machine architecture — [source](#)

And [here](#) is a paper that applies Transformer language models to Extractive and Abstractive Neural document summarization.

Researchers developed a method that focuses on using comparisons to build and train ML models. Instead of requiring large amounts of feature-label pairs, this [technique](#) compares images with previously seen images to decide whether the image should be of a certain label.

Nelson Liu and others presented a [paper](#) discussing the type of linguistic knowledge being captured by pretrained contextualizers such as BERT and ELMo.

[XLNet](#) is a pretraining method for NLP that showed improvements upon BERT on 20 tasks. I wrote a summary of this great work [here](#).

This [work](#) from DeepMind reports the results from an extensive empirical investigation that aims to evaluate language understanding models applied to a variety of tasks. Such extensive analysis is important to better understand what language models capture so as to improve their efficiency.

[VisualBERT](#) is a simple and robust framework for modeling vision-and-language tasks including VQA and Flickr30K, among others. This approach leverages a stack of Transformer layers coupled with self-attention to align elements in a piece of text and the regions of an image.

This [work](#) provides a detailed analysis comparing NLP transfer learning methods along with guidelines for NLP practitioners.

Alex Wang and Kyunghyun propose an [implementation of BERT](#) that is able to produce high-quality, fluent generations. Here is a Colab [notebook](#) to try it.

Facebook researchers published code ([PyTorch implementation](#)) for XLM which is a model for cross-lingual model pretraining.

This [works](#) provides a comprehensive analysis of the application of reinforcement learning algorithms for neural machine translation.

This survey [paper](#) published in JAIR provides a comprehensive overview of the training, evaluation, and use of cross-lingual word embedding models.

The Gradient published an excellent [article](#) detailing the current limitations of reinforcement learning and also providing a potential path forward with hierarchical reinforcement learning. And in a timely manner, a couple of folks published an excellent set of [tutorials](#) to get started with reinforcement learning.

This [paper](#) provides a light introduction to contextual word representations.

## ML/NLP Creativity and Society

*Machine learning has been applied to solve real-world problems but it has also been applied in interesting and creative ways. ML creativity is as important as any other research area in AI because at the end of the day we wish to build AI systems that will help shape our culture and society.*

Towards the end of this year, Gary Marcus and Yoshua Bengio [debated](#) on the topics of deep learning, symbolic AI and the idea of hybrid AI systems.

The [2019 AI Index Report](#) was finally released and provides a comprehensive analysis of the state of AI which can be used to better understand the progress of AI in general.

[Commonsense reasoning](#) continues to be an important area of research as we aim to build artificial intelligence systems that not only are able to make a prediction on the data provided but also understand and can reason about those decisions. This type of technology can be used in conversational AI where the goal is to enable an intelligent agent to have more natural conversations with people. Check out this [interview](#) with Nasrin Mostafazadeh having a discussion on commonsense reasoning and applications such as storytelling and language understanding. You can also check out this recent [paper](#) on how to leverage language models for commonsense reasoning.

[Activation Atlases](#) is a technique developed by researchers at Google and Open AI to better understand and visualize the interactions happening between neurons of a neural network.





*“An activation atlas of the InceptionV1 vision classification network reveals many fully realized features, such as electronics, buildings, food, animal ears, plants, and watery backgrounds.” — [source](#)*

Check out the [Turing Lecture](#) delivered by Geoffrey Hinton and Yann LeCun who were [awarded](#), together with Yoshua Bengio, the Turing Award this year.

Tackling climate change with machine learning is discussed in this [paper](#).

OpenAI published an extensive [report](#) discussing the social impacts of language models covering topics like beneficial use and potential misuse of the technology.

Emotion analysis continues to be used in a diverse range of applications. [The Mojifier](#) is a cool project that looks at an image, detects the emotion, and replaces the face with the emojis matching the emotion detected.

Work on radiology with the use of AI techniques has also been trending this year. Here is a nice [summary](#) of trends and perspectives in this area of study. Researchers from



NYU also released a [Pytorch implementation](#) of a deep neural network that improves radiologists' performance on breast cancer screening. And here is a major [dataset](#) release called MIMIC-CXR which consists of a database of chest Xrays and text radiology reports.

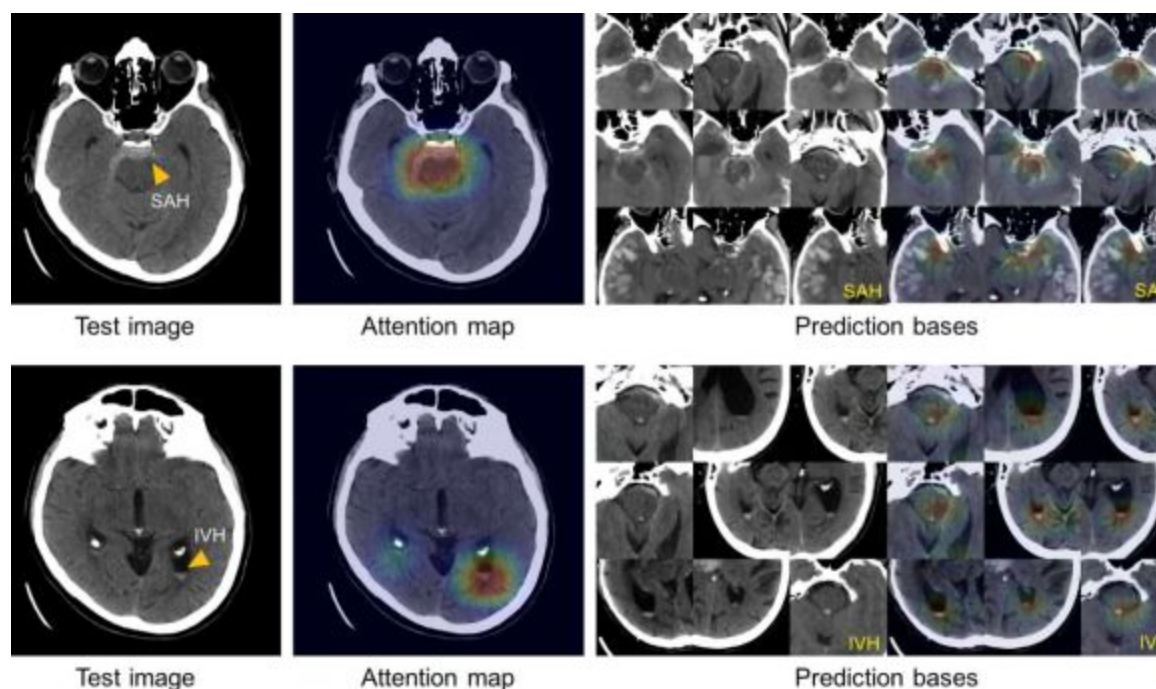
The New York Times wrote a [piece](#) on Karen Spark Jones remembering the seminal contributions she made to NLP and Information Retrieval.

OpenAI Five [became](#) the first AI system to beat a world champion at an esports game.

The [Global AI Talent Report](#) provides a detailed report of the worldwide AI talent pool and the demand for AI talent globally.

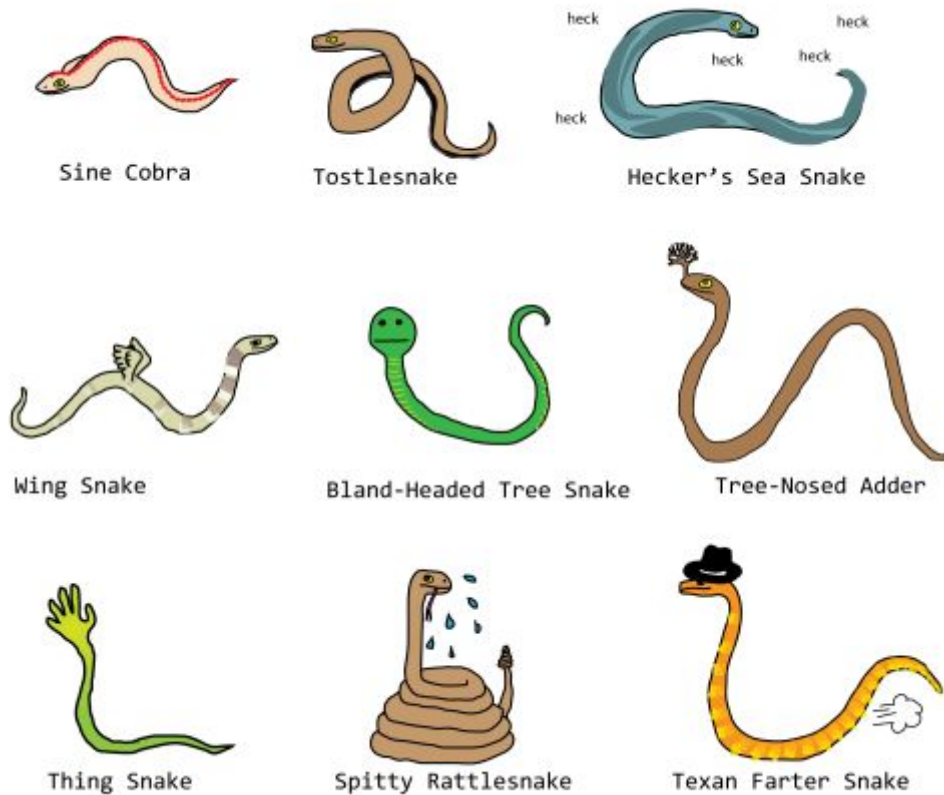
If you haven't subscribed already, the DeepMind team has an excellent [podcast](#) where participants discuss the most pressing topics involving AI. Talking about AI potential, Demis Hassabis did an [interview](#) with The Economist where he spoke about futuristic ideas such as using AI as an extension to the human mind to potentially find solutions to important scientific problems.

This year also witnessed incredible advancement in ML for health applications. For instance, researchers at Massachusetts [developed](#) an AI system capable of spotting brain hemorrhages as accurate as humans.



*"Brain scans analyzed by the AI system."*

Janelle Shane summarizes a set of [“weird” experiments](#) showing how machine learning can be used in creative ways to conduct fun experimentation. Sometimes this is the type of experiment that’s needed to really understand what an AI system is actually doing and not doing. Some experiments include neural networks generating fake snakes and telling jokes.



### [Snake Species](#)

[Learn](#) to find planets with machine learning models build on top of TensorFlow.

OpenAI [discusses](#) the implication of releasing (including the potential of malicious use cases) large-scale unsupervised language models.

This [Colab notebook](#) provides a great introduction on how to use Nucleus and TensorFlow for “DNA Sequencing Error Correction”. And here is a great detailed [post](#) on the use of deep learning architectures for exploring DNA.

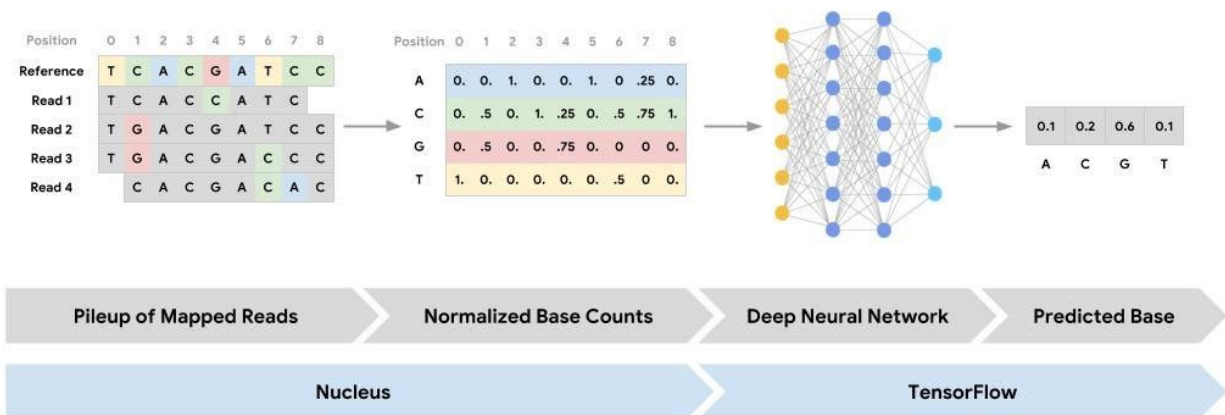


Figure 1: We formulate consensus-based DNA sequencing error correction as a multiclass classification problem. Using Nucleus, we construct a matrix of normalized base counts in a genomic range. TensorFlow allows us to train a neural network that can predict the correct base at the middle position of the window.

[source](#)

Alexander Rush is a Harvard NLP researcher who wrote an important article about the [issues](#) with tensors and how some current libraries expose them. He also went on to talk about a proposal for tensors with named indices.

## ML/NLP Tools and Datasets

*Here I highlight stories related to software and datasets that have assisted in enabling NLP and machine learning research and engineering.*

Hugging Face released a popular Transformer [library](#) based on Pytorch names pytorch-transformers. It allows NLP practitioners and researchers to easily use state-of-the-art general-purpose architectures such as BERT, GPT-2, and XLM, among others. If you are interested in how to use pytorch-transformers there are a few places to start but I really liked this detailed [tutorial](#) by Roberto Silveira showing how to use the library for machine comprehension.

# Transformers

[Source](#)

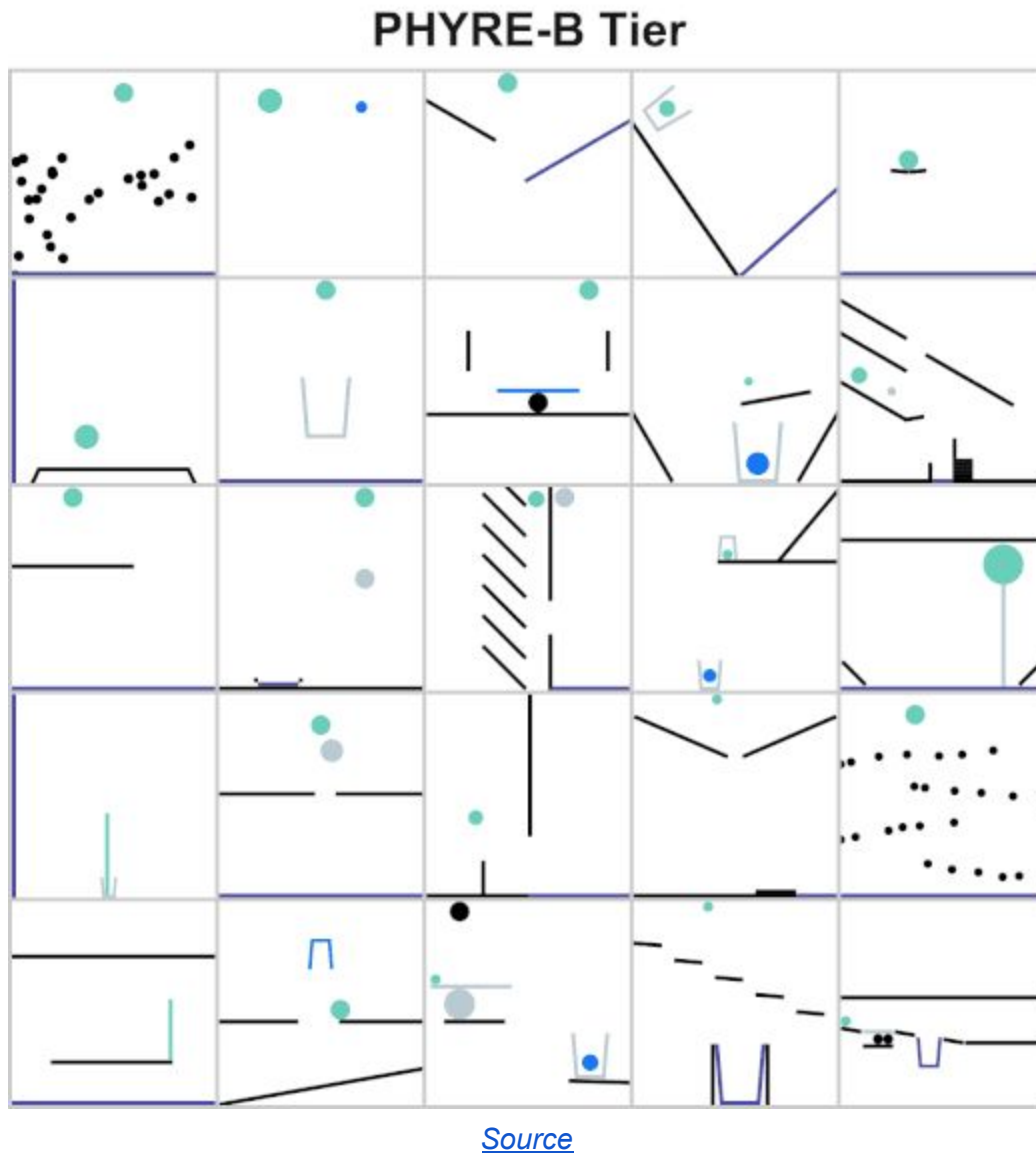
TensorFlow 2.0 was released with a bunch of [new features](#). Read more about best practices [here](#). François Chollet also wrote an extensive overview of the new features in this [Colab notebook](#).

PyTorch 1.3 was [released](#) with a ton of new features including named tensors and other front-end improvements.

The Allen Institute for AI released [Iconary](#) which is an AI system that can play Pictionary-style games with a human. This work incorporates visual/language learning systems and commonsense reasoning. They also published a new commonsense reasoning [benchmark](#) called Abductive-NLI.

spaCy [releases](#) a new library to incorporate Transformer language models into their own library so as to be able to extract features and used them in spaCy NLP pipelines. This effort is built on top of the popular Transformers library developed by Hugging Face. Maximilien Roberti also [wrote](#) a nice article on how to combine fast.ai code with pytorch-transformers.

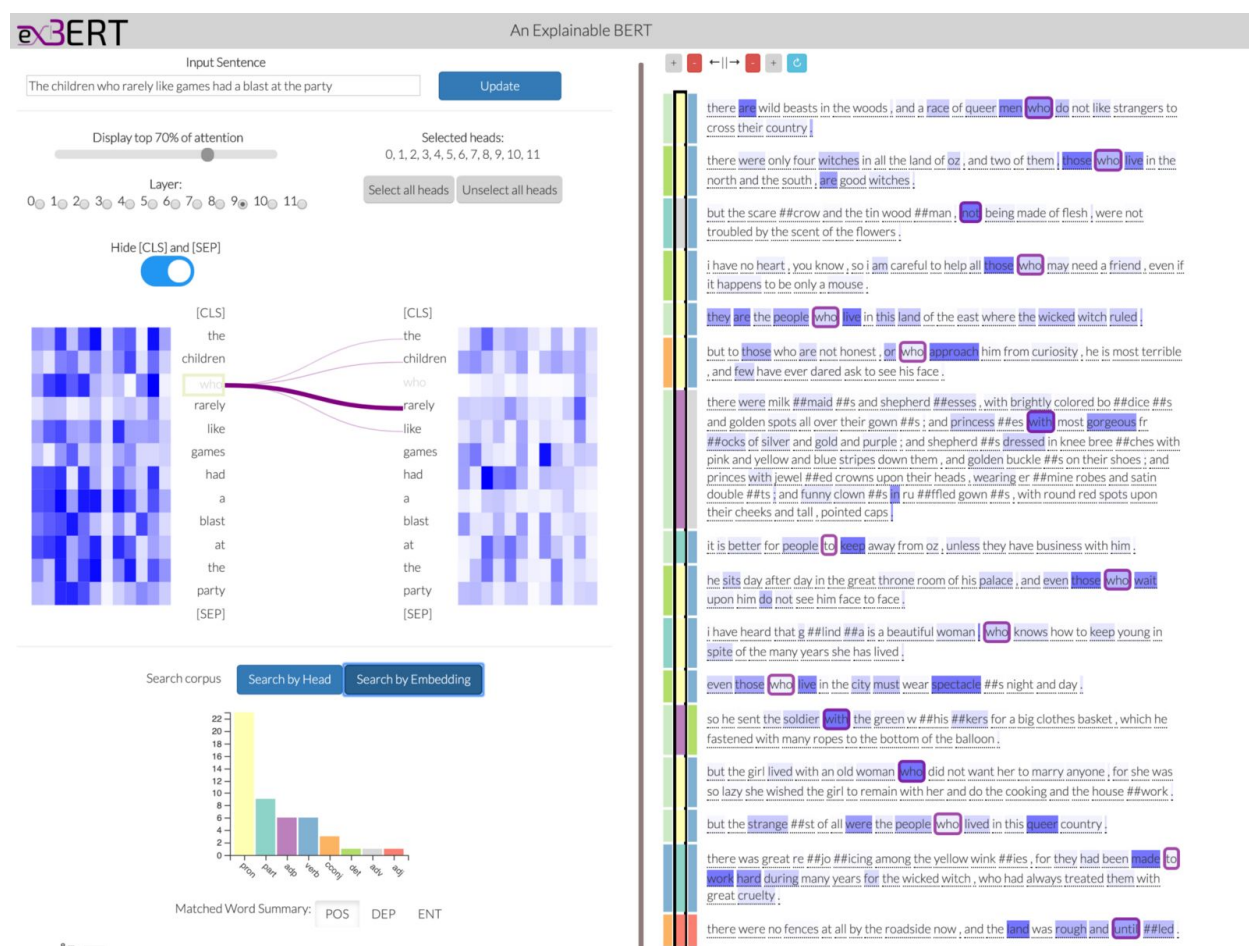
The Facebook AI team released [PHYRE](#) which is a benchmark for physical reasoning aiming to test the physical reasoning of AI systems through solving various physics puzzles.



StanfordNLP released [StanfordNLP 0.2.0](#) which is a Python library for natural language analysis. You can perform different types of linguistic analysis such as lemmatization and part of speech recognition on over 70 different languages.

[GQA](#) is a visual question answering dataset for enabling research related to visual reasoning.

exBERT is a visual interactive tool to explore the embeddings and attention of Transformer language models. You can find the paper [here](#) and the demo [here](#).



exBERT — [source](#)

Distill published an [article](#) on how to visualize memorization in Recurrent Neural Networks (RNNs).

[Mathpix](#) is a tool that lets you take a picture of an equation and then it provides you with the latex version.

[Parl.ai](#) is a platform that hosts many popular datasets for all works involving dialog and conversational AI.

Uber researchers released [Ludwig](#), an open-source tool that allows users to easily train and test deep learning models with just a few lines of codes. The whole idea is to avoid any coding while training and testing models.



Google AI researchers release “[Natural Questions](#)” which is a large-scale corpus for training and evaluating open-domain question answering systems.

## Articles and Blog posts

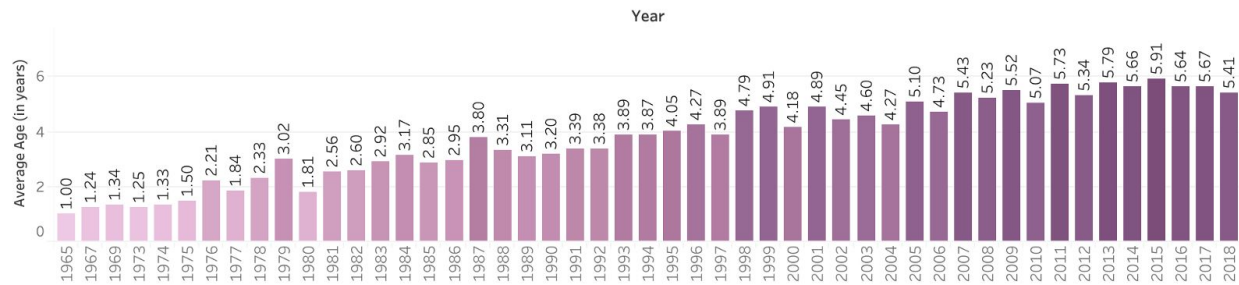
*This year witnessed an explosion of data science writers and enthusiasts. This is great for our field and encourages healthy discussion and learning. Here I list a few interesting and must-see articles and blog posts I came across:*

Christian Perone provides an [excellent introduction](#) to maximum likelihood estimation (MLE) and maximum a posteriori (MAP) which are important principles to understand how parameters of a model are estimated.

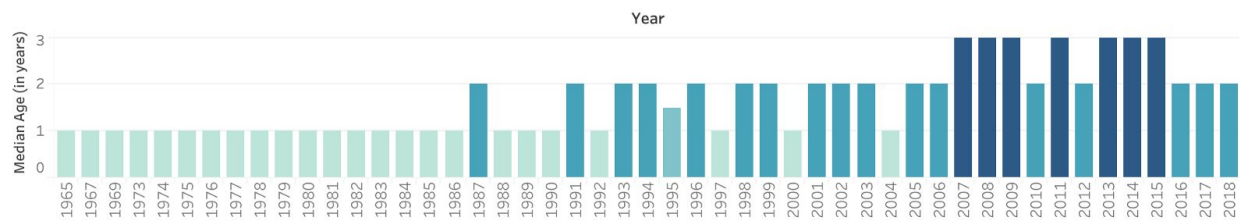
Reiichiro Nakano published a [blog post](#) discussing neural style transfer with adversarially robust classifiers. A Colab [notebook](#) was also provided.

Saif M. Mohammad started a great [series](#) discussing a diachronic analysis of ACL anthology.

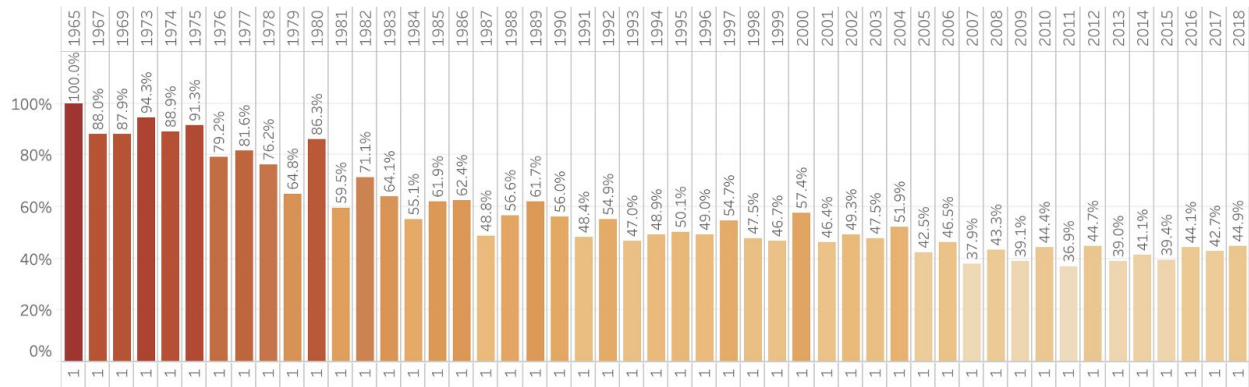
### Average Academic Age of NLP Researchers



### Median Academic Age of NLP Researchers



### Percentage of authors who published their first paper in AA (NLP Academic Age 1)



*“Graphs showing average academic age, median academic age, and percentage of first-time publishers in AA over time.” — [source](#)*

The question is: can a language model learn syntax? Using structural probes, this [work](#) aims to show that it is possible to do so using contextualized representations and a method for finding tree structures.

Andrej Karpathy wrote a [blog post](#) summarizing best practices and a recipe on how to effectively train neural networks.

Google AI researchers and other researchers collaborated to [improve](#) the understanding of search using BERT models. Contextualized approaches like BERT are adequate to understand the intent behind search queries.

[Rectified Adam](#) (RAdam) is a new optimization technique based on Adam optimizer that helps to improve AI architectures. There are several efforts in coming up with better and more stable optimizers but the authors claim to focus on other aspects of optimizations that are just as important for delivering improved convergence.

With a lot of development of machine learning tools recently, there are also many discussions on how to implement ML systems that enable solutions to practical problems. Chip Huyen [wrote](#) an interesting chapter discussing machine learning system design emphasizing on topics such as hyperparameter tuning and data pipeline.

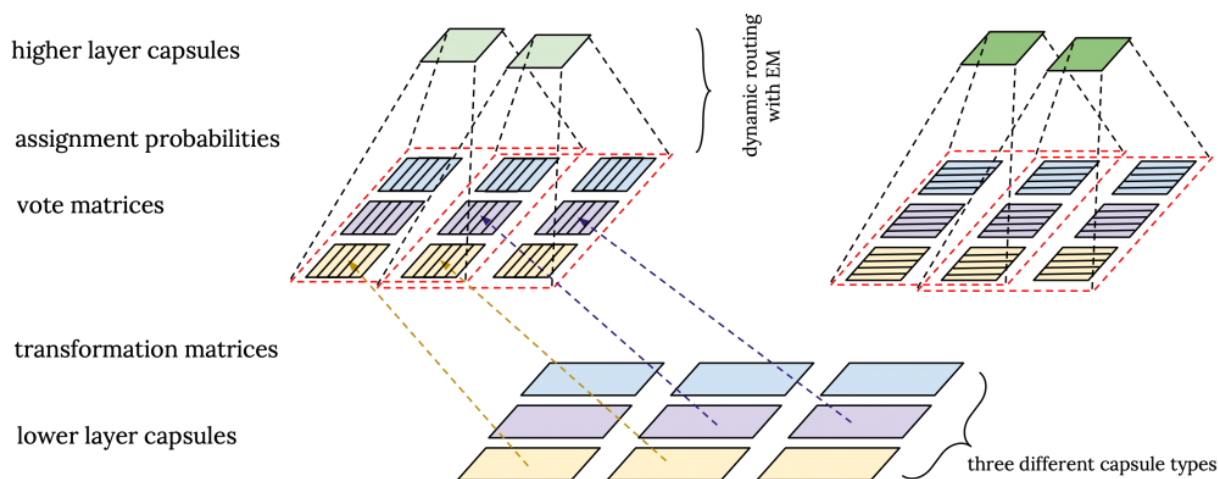
NVIDIA [breaks the record](#) for creating the biggest language model trained on billions of parameters.

Abigail See wrote this excellent blog [post](#) about what makes a good conversation in the context of systems developed to perform natural language generation task.

Google AI [published](#) two natural language dialog datasets with the idea to use more complex and natural dialog datasets to improve personalization in conversational applications like digital assistants.

Deep reinforcement learning continues to be one of the most widely discussed topics in the field of AI and it has even attracted interest in the space of psychology and neuroscience. Read more about some highlights in this [paper](#) published in Trends in Cognitive Sciences.

Samira Abner wrote this excellent [blog post](#) summarizing the main building blocks behind Transformers and capsule networks and their connections. Adam Kosiorek also wrote this magnificent [piece](#) on stacked capsule-based autoencoders (an unsupervised version of capsule networks) which was used for object detection.



*Connection between two adjacent capsule layers, where the lower layer has three capsule types and the higher layer has two capsule types.*

[source](#)

Researchers published an [interactive article](#) on Distill that aims to show a visual exploration of [Gaussian Processes](#).

Through this Distill [publication](#), Augustus Odena makes a call to researchers to address several important open questions about GANs.

Here is a PyTorch [implementation](#) of graph convolutional networks (GCNs) used for classifying spammers vs. non-spammers.

At the beginning of the year, VentureBeat released a list of [predictions](#) for 2019 made by experts such as Rumman Chowdury, Hilary Mason, Andrew Ng, and Yan LeCun. Check it out to see if their predictions were right.

[Learn](#) how to finetune BERT to perform multi-label text classification.

Due to the popularity of BERT, in the past few months, many researchers developed methods to “compress” BERT with the idea to build faster, smaller and memory-efficient

versions of the original. Mitchell A. Gordon [wrote](#) a summary of the types of compressions and methods developed around this objective.

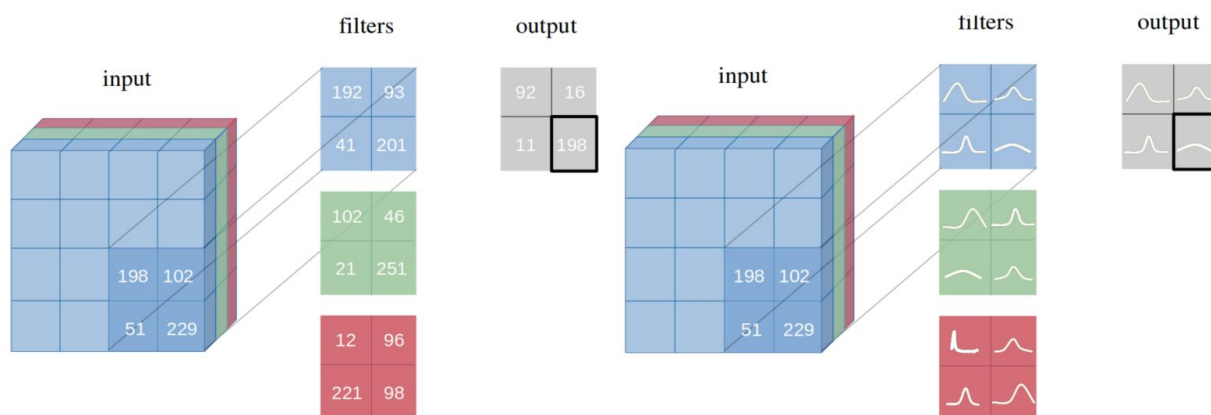
*Superintelligence* continued to be a topic of debate among experts. It's an important topic that needs a proper understanding of frameworks, policies, and careful observations. I found this interesting [series of comprehensive essays](#) (in the form of a technical report by K. Eric Drexler) to be useful to understand some issues and considerations around the topic of superintelligence.

Eric Jang wrote a nice [blog post](#) introducing the concept of meta-learning which aims to build and train machine learning models that not only predict well but also learn well.

A [summary](#) of AAAI 2019 highlights by Sebastian Ruder.

Graph neural networks were heavily discussed this year. David Mack wrote a [nice visual article](#) about how they used this technique together with attention to perform shortest path calculations.

Bayesian approaches remain an interesting subject, in particular how they can be applied to neural networks to avoid common issues like over-fitting. Here is a [list](#) of suggested reads by Kumar Shridhar on the topic.



*“Network with point-estimates as weights vs Network with probability distribution as weights” - [Source](#)*



## Ethics in AI

*Perhaps one of the most highly discussed aspects of AI systems this year was ethics which include discussions around bias, fairness, and transparency, among others. In this section, I provide a list of interesting stories and papers around this topic:*

The paper titled “[Does mitigating ML’s impact disparity require treatment disparity?](#)” discusses the consequences of applying disparate learning processes through experiments conducted on real-world datasets.

HuggingFace published an [article](#) discussing ethics in the context of open-sourcing NLP technology for conversational AI.

This [work](#) presented at NAACL 2019 discusses how debiasing methods can cover up gender bias in word embeddings.

[Listen](#) to Zachary Lipton presenting his paper “Troubling Trends in ML Scholarship”. I also wrote a summary of this interesting paper which you can find [here](#).

Gary Marcus and Ernest Davis published their book on “[Rebooting AI: Building Artificial Intelligence We Can Trust](#)”. The main theme of the book is to talk about the steps we must take to achieve robust artificial intelligence. On the topic of AI progression, François Chollet also wrote an impressive [paper](#) making a case for better ways to measure intelligence.

Check out this Udacity [course](#) created by Andrew Trask on topics such as differential privacy, federated learning, and encrypted AI. On the topic of privacy, Emma Bluemke wrote this great [post](#) discussing how one may go about training machine learning models while preserving patient privacy.

At the beginning of this year, Mariya Yao [posted](#) a comprehensive list of research paper summaries involving AI ethics. Although the list of paper reference was from 2018, I believe they are still relevant today.

## ML/NLP Education

*Here I will feature a list of educational resources, writers and people doing some amazing work educating others about difficult ML/NLP concepts/topics:*

CMU released materials and syllabus for their “[Neural Networks for NLP](#)” course.

Microsoft Research Lab published a free [ebook](#) on the foundation of data science with topics ranging from Markov Chain Monte Carlo to Random Graphs.

“[Mathematics for Machine Learning](#)” is a free ebook introducing the most important mathematical concepts used in machine learning. It also includes a few Jupyter notebook tutorials describing the machine learning parts. Jean Gallier and Jocelyn Quaintance wrote an [extensive free ebook](#) covering mathematical concepts used in machine learning.

Stanford releases a [playlist of videos](#) for its course on “Natural Language Understanding”.

On the topic of learning, OpenAI put together this great [list](#) of suggestions on how to keep learning and improving your machine learning skills. Apparently, their employees use these methods on a daily basis to keep learning and expanding their knowledge.

## Learning day is a gift ❤️

Feel free to use learning day for:

- Reading AI papers
- Reimplementing AI papers
- Going over AI tutorials
- Having your tiny side research AI projects
- Learning about fundamentals: linear algebra, statistics etc.
- Learning about ML fundamentals: information theory, Bayesian inference etc.
- Learning about engineering. Read about new programming languages, frameworks (e.g. what is Rust?)
- Learning about management: self-management, prioritization, how renowned research labs works (e.g. Manhattan project, Bell labs)
- Learning about mechanical engineering or having your mechanical engineering research project (e.g. build a linear actuator)
- Learning about any skill that will level you up in your work (in my case, it's better writing)

[source](#)

Adrian Rosebrock [published](#) an 81-page guide on how to do computer vision with Python and OpenCV.

Emily M. Bender and Alex Lascarides published a [book](#) titled “Linguistic Fundamentals for NLP”. The main idea behind the book is to discuss what

“meaning” is in the field of NLP by providing a proper foundation on semantics and pragmatics.

Elad Hazan published his [lecture notes](#) on “Optimization for Machine Learning” which aims to present machine training as an optimization problem with beautiful math and notations. Deeplearning.ai also published a great [article](#) that discusses parameter optimization in neural networks using a more visual and interactive approach.

Andreas Mueller published a [playlist](#) of videos for a new course in “Applied Machine Learning”.

Fast.ai [releases](#) its new MOOC titled “Deep Learning from the Foundations”.

MIT published all [videos](#) and syllabus for their course on “Introduction to Deep Learning”.

Chip Huyen [tweeted](#) an impressive list of free online courses to get started with machine learning.

Andrew Trask [published](#) his book titled “Grokking Deep Learning”. The book serves as a great starter for understanding the fundamental building blocks of neural network architectures.

Sebastian Raschka uploaded [80 notebooks](#) about how to implement different deep learning models such as RNNs and CNNs. The great thing is that the models are all implemented in both PyTorch and TensorFlow.

Here is a great [tutorial](#) that goes deep into understanding how TensorFlow works. And here is [one](#) by Christian Perone for PyTorch.

Fast.ai also published a course titled “Intro to NLP” accompanied by a [playlist](#). Topics range from sentiment analysis to topic modeling to the Transformer.

[Learn](#) about Graph Convolutional Neural Networks for Molecular Generation in this talk by Xavier Bresson. Slides can be found [here](#). And here is a paper discussing how to [pre-train](#) GNNs.

On the topic of graph networks, some engineers [use them](#) to predict the properties of molecules and crystal. The Google AI team also published an excellent blog [post](#) explaining how they use GNNs for odor prediction. If you are interested in getting started with Graph Neural Networks, here is a comprehensive [overview](#) of the different GNNs and their applications.

Here is a [playlist](#) of videos on unsupervised learning methods such as PCA by Rene Vidal from Johns Hopkins University.

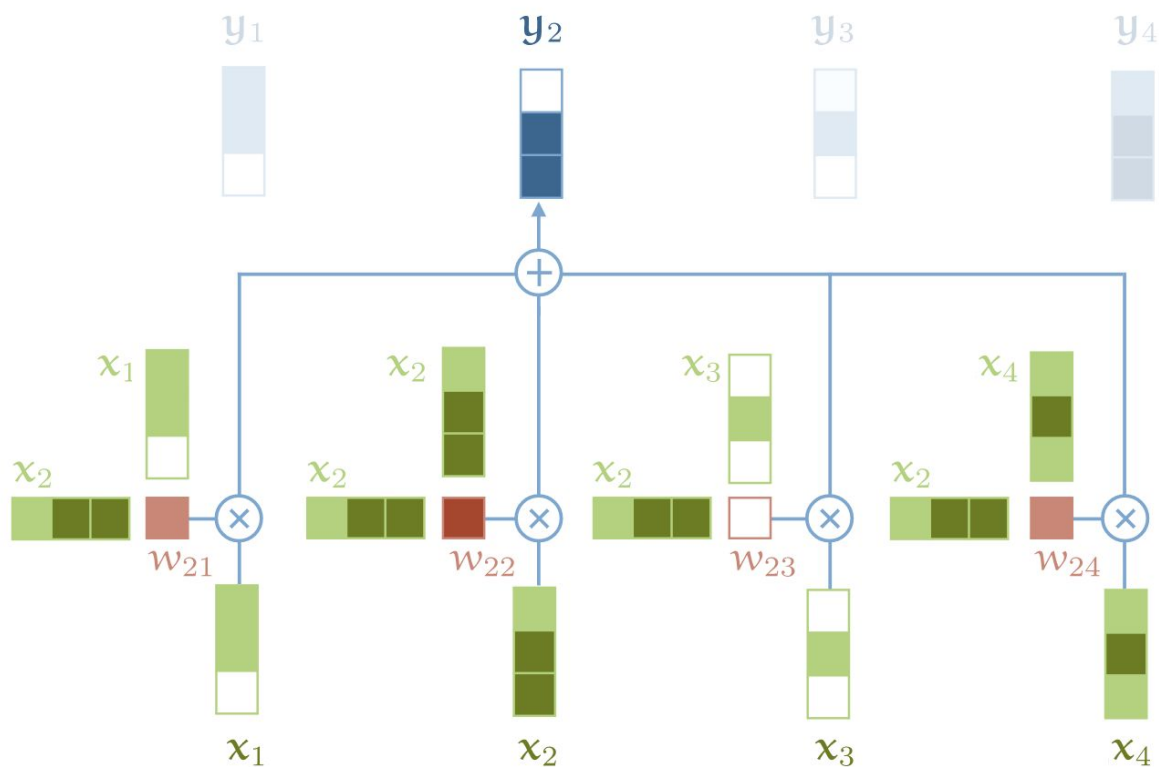
If you are ever interested in converting a pretrained TensorFlow model to PyTorch, Thomas Wolf has you covered in this [blog post](#).

Want to learn about generative deep learning? David Foster wrote a great [book](#) that teaches data scientists how to apply GANs and encoder-decoder models for performing tasks such as patining, writing, and composing music. Here is the official [repository](#) accompanying the book, it includes TensorFlow code. There is also an [effort](#) to convert the code to PyTorch as well.

A Colab [notebook](#) containing code blocks to practice and learn about causal inference concepts such as interventions, counterfactuals, etc.

Here are the [materials](#) for the NAACL 2019 tutorial on “Transfer Learning in Natural Language Processing” delivered by Sebastian Ruder, Matthew Peters, Swabha Swayamdipta and Thomas Wolf. They also provided an accompanying Google Colab [notebook](#) to get started.

Another great [blog post](#) from Jay Alammar on the topic of data representation. He also wrote many other interesting illustrated guides that include [GPT-2](#) and [BERT](#). Peter Bloem also published a very detailed [blog post](#) explaining all the bits that make up a Transformer.



A visual illustration of basic self-attention — [source](#)

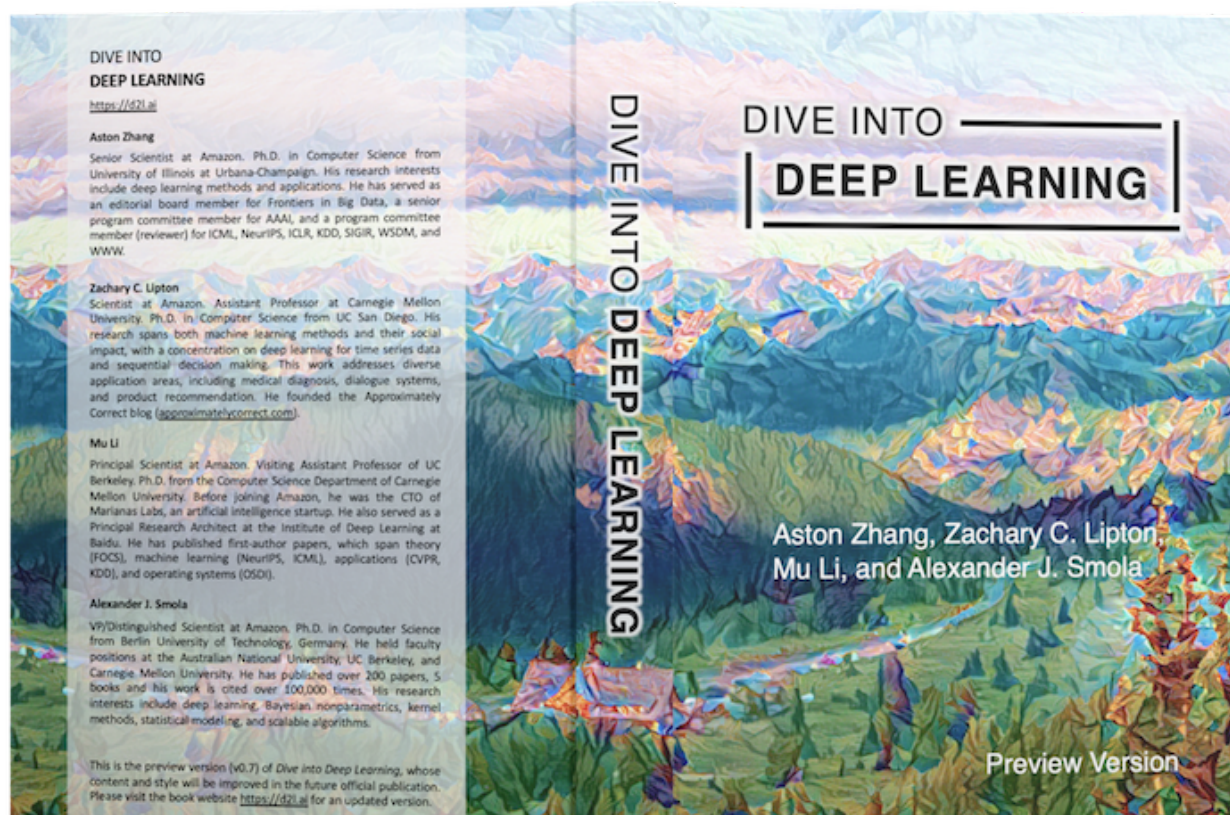
Here is a nice overview of trends in NLP at ACL 2019, written by [Mihail Eric](#). Some topics include infusing knowledge into NLP architectures, interpretability, and reducing bias among others. Here are a couple more overviews if you are interested: [link 2](#) and [link 3](#).

The full syllabus for CS231n 2019 edition was [released](#) by Stanford.

David Abel [posted](#) a set of notes for ICLR 2019. He was also nice to provide an impressive [summary](#) of NeurIPS 2019.

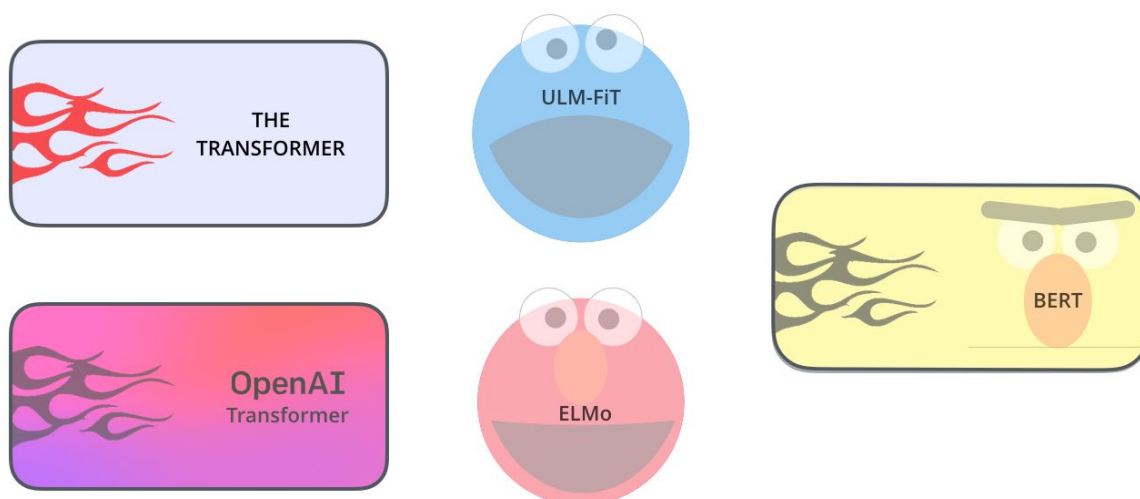
This is an excellent [book](#) that provides learners with a proper introduction to deep learning with notebooks provided as well.





[source](#)

An illustrated [guide](#) to BERT, ELMo, and co. for transfer learning NLP.



[source](#)

Fast.ai [releases](#) its 2019 edition of the “Practical Deep Learning for Coders” course.

[Learn](#) about deep unsupervised learning in this fantastic course taught by Pieter Abbeel and others.

Gilbert Strang [released](#) a new book related to Linear Algebra and neural networks.

Caltech provided the entire syllabus, lecture slides, and video playlist for their course on “[Foundation of Machine Learning](#)”.

The “[Scipy Lecture Notes](#)” is a series of tutorials that teach you how to master tools such as matplotlib, NumPy, and SciPy.

Here is an excellent [tutorial](#) on understanding Gaussian processes. (Notebooks provided).

This is a must-read [article](#) in which Lilian Weng provides a deep dive into generalized language models such as ULMFit, OpenAI GPT-2, and BERT.

[Papers with Code](#) is a website that shows a curated list of machine learning papers with code and state-of-the-art results.

Christoph Molnar released the first edition of “[Interpretable Machine Learning](#)” which is a book that touches on important techniques used to better interpret machine learning algorithms.

David Bamman [releases](#) the full syllabus and slides to the NLP courses offered at UC Berkley.

Berkley [releases](#) all materials for their “Applied NLP” class.

Aerin Kim is a senior research engineer at Microsoft and [writes](#) about topics related to applied Math and deep learning. Some topics include intuition to conditional independence, gamma distribution, perplexity, etc.

Tai-Danae Bradley wrote this [blog post](#) discussing ways on how to think about matrices and tensors. The article is written with some incredible visuals which help to better understand certain transformations and operations performed on matrices.

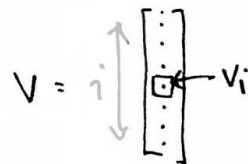
a number



a vector



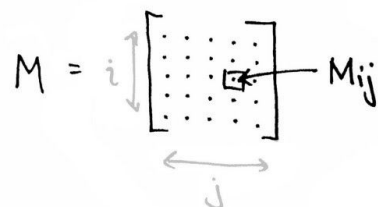
$v_i$



a matrix



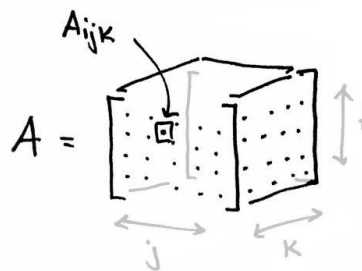
$M_{ij}$



a 3-tensor



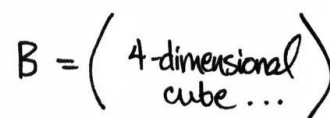
$A_{ijk}$



a 4-tensor



$B_{ijkl}$



$\vdots$

$\vdots$

$\vdots$

$\vdots$

[source](#)