INL/EXT-21-61935
Revision 0

An Introduction to Word Embeddings and Language Models

April 2021

Tammie Borders
Idaho National Laboratory

Svitlana Volkova
Pacific Northwest National Laboratory

Idaho National Laboratory

INL is a U.S. Department of Energy National Laboratory
operated by Batelle Energy Alliance, LLC

INL/
Revision 0

An Introduction to Word Embeddings and Language Models

Tammie Borders
Idaho National Laboratory

Svitlana Volkova
Pacific Northwest National Laboratory

April 2021

Idaho National Laboratory
Idaho Falls, Idaho 83415

http://www.inl.gov

Prepared for the
U.S. Department of Energy
National Nuclear Security Administration
Defense Nuclear Nonproliferation R&D
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

Page intentionally left blank

An Introduction to Word Embeddings and Language Models

1. Introduction

Language models have advanced at a phenomenal pace over the past decade [1]. This document provides a short introduction to terminology, word embeddings (aka low-dimensional representations), and popular large-scale language models (LMs). Word embeddings are used to represent words as numerical vectors and are context-independent, meaning a word can only have a single representation (e.g., *club* can only be *club* sandwich, not golf *club*). Language models can determine the probability of a given sequence of words occurring in a sentence and can provide context to distinguish between words and phrases that sound similar. LMs are context-dependent (e.g., *club* can be *club* sandwich or golf *club*) and largely fall in two main classes – autoregressive and autoencoding models.

Autoregressive models are pretrained on the classic language modeling task: guess the next token having read all the previous ones. Those models can be fine-tuned and achieve great results on many tasks, the most natural application is text generation. A typical example of such models is GPT, but others include GPT-2, GPT-3, CTLR, TRANSFORMER-XL, REFORMER, XLNET.

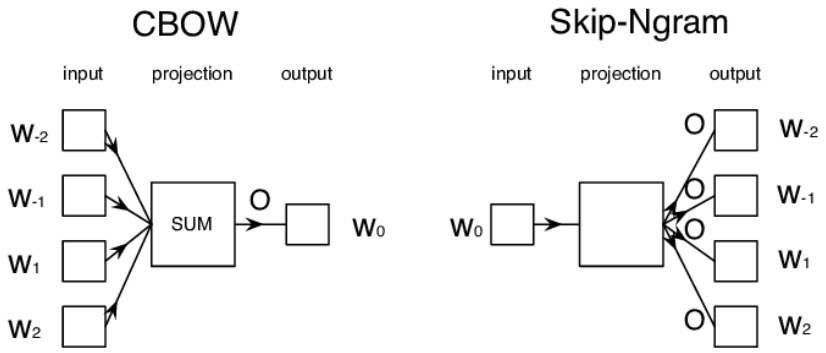
Autoencoding models are pretrained by corrupting the input tokens in some way and trying to reconstruct the original sentence. They can be fine-tuned and achieve great results on many tasks such as text generation, but their most natural application is sentence classification or token classification. A typical example of such models is BERT, but others include ROBERTA, ALBERT, XML, XML-ROBERTA, FLAUBERT AND LONGFORMER.

2. Pretrained Word Embedding Models

Language is an extremely high-dimensional space to operate in and encode for modeling purposes. Word embeddings are a popular representation of language through learning vector representations of a specific word (or a phrase, a sentence, or the whole document), also known as text data vectorization. These vector representations are used to facilitate association of context to words numerically. The first pre-trained, widespread word embeddings model was WORD2VEC in 2013 [2]. Other popular models include GLOVE, FASTTEXT [3, 4].

Word embeddings capture word context relative to other words and are used to generate predictions of nearby words. They are key to text analysis in natural language processing (NLP) tasks, such as sentiment analysis, topic extraction, topic classification. A key limitation of a word embedding is that it is context independent. For example, a word embedding can only have a single context representation for the word *club*, whereas *club* could mean *golf club, clubhouse, club sandwich* [5-7]. The simplest method of text data vectorization, in the pre-word embedding learning era, is one-hot encoding. The one-hot encoding for "brown" and "fast", as the second and fifth words in the sentence "the brown dog ran fast" would be [0,1,0,0,0] and [0,0,0,0,1].

Two popular architectures for learning word embeddings are skip-gram and continuous bag of words (CBOW) [8]. Each loop on words in a text corpus and makes predictions. Skip-gram uses the current word to predict its neighboring words while CBOW uses its neighboring words to predict the current word. Skip-gram works well with a small amount of training data and instances of rare words while CBOW is much faster to train with better performance for frequent words. To encode context information, ELMO [7] embeddings emerged in 2018 and have been widely used to initialize NLP models.



3. Autoencoding Language Models: BERT and Transformer Models

Language models can determine the probability of a given sequence of words occurring in a sentence and can provide context to distinguish between words and phrases that sound similar. WORD2VEC is context independent while language models are context dependent. Language models can distinguish between *golf club and club sandwich* based on the context of the sentence.

The Bidirectional Encoder Representations from Transformers (BERT) language model was released in 2017 [9, 10]. Bidirectional means contextual information about a word is learned simultaneously from both left-to-right and right-to-left directions. Until transformers, language models were uni-directional. Key observations about the BERT model:

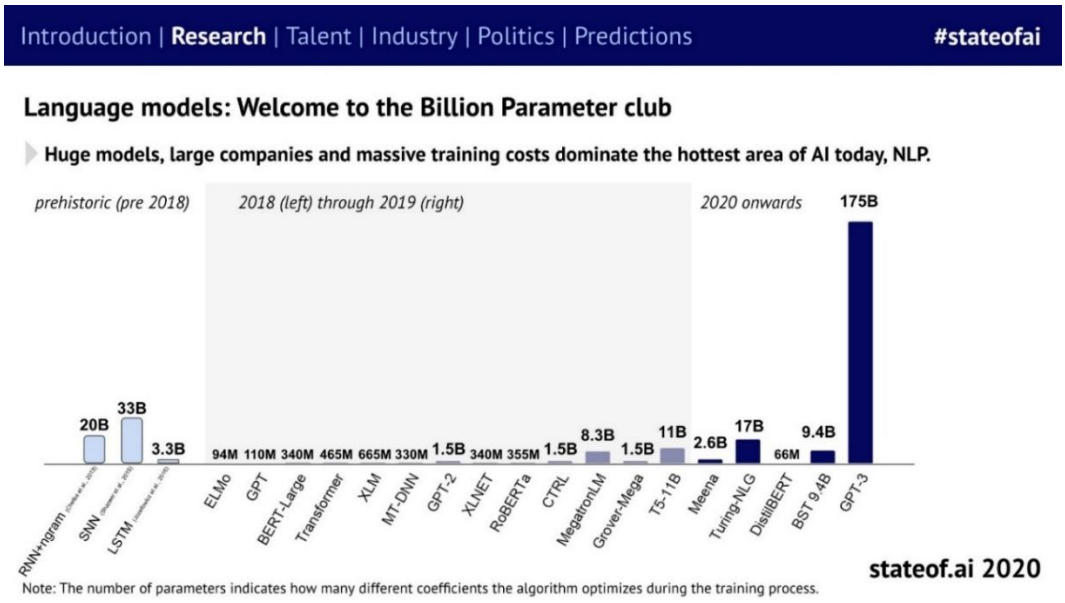
- Model size is critical, more parameters equal superior model accuracy.
- More training steps with more training data correlates to higher model accuracy.
- BERT's bidirectional approach converges slower than left-to-right.

Several terms are key to understanding language models. *Pretrained* means the language model has been trained on a large text corpus and can understand the language; in effect, creating a well-read person. *Fine-tuning* a model is used when the model needs to represent a specialized dataset (e.g., BioBERT). *General purpose* models, like BERT and GPT-3, are models trained with large numbers of parameters (millions to billions) and are expected to perform close to fine-tuned models.

4. Autoregressive Models: GPT-3 and Next-Gen Transformers

In May 2020, OpenAI released a paper on Generative Pre-trained Transformer 3 (GPT-3) with 175 billion parameters, making it the largest transformer-based language model in the world at the time [11]. The following chart shows the relative size of GPT-3 versus prior language models; note, the largest BERT architecture (BERT-Large) has 340 million parameters [12].

In contrast to BERT, GPT-3 does not require substantial fine tuning; however, there are a lack of studies for specialized datasets (e.g., nonproliferation). GPT-3 generates output with *few-shot learning*, *one-shot*, and *zero-shot learning* [13]. *Few-shot learning* is defined as fine tuning with few data examples, *one-shot* is fine tuning with one example, and *zero-shot* is no fine-tuning at all. GPT-3 demonstrated an unexpected proficiency across a range of tasks, including software code generation. It is believed GPT-3 will work well on multiple non-text data types. GPT-3 is not available open source. DALL-E, a recent GPT-3 variant, can draw images from text descriptions [14]. Other multimodal models include CLIP [15], and UniT [16].



5. Switch Transformer

Google announced a 1.6 trillion parameter model in January 2021, based on the switch transformer architecture [15-17]. The architecture is more computationally efficient that BERT or GPT-3 and initial benchmarking performance data shows promise, stay tuned for further information.

6. Comparison: BERT, GPT-3

The following table summarizes a comparison of BERT versus GPT-3.

Attribute	Autoencoding: BERT	Autoregressive: GPT-3
Number of Parameters	340 Million	175 Billion
Required Fine Tuning	Substantial	Few- to Zero-Shot Learning
Access Availability	Open Source	Commercial
Run Model Size (Memory)	1.4 GB	700 GB
Cost	• Free Access • Moderate Labor • Moderate Compute	• Pay for Use • Minimal Labor Required • High-End Compute
Capability (Language Tasks)	Very Good Requires fine tuning	Excellent May not require fine tuning
Capability (Beyond Languages)	No	Images (DAL-E), Software, Unknown

7. References

1. Tan, T., Evolution of Language Models: N-Grams, Word Embeddings, Attention & Transformers, in Towards Data Science. 2020.
2. Mikolov, T., et al. Distributed Representations of Words and Phrases and their Compositionality. 2013. arXiv:1310.4546.
3. Bojanowski, P., et al. Enriching Word Vectors with Subword Information. 2016. arXiv:1607.04606.
4. Pennington, J., R. Socher, and C. Manning. GloVe: Global Vectors for Word Representation. in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014. Doha, Qatar: Association for Computational Linguistics.
5. Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K. and Zettlemoyer, L., 2018, June. Deep Contextualized Word Representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers) (pp. 2227-2237).
6. Kulshrestha, R., NLP 101: Word2Vec — Skip-gram and CBOW, in Towards Data Science. 2019.
7. Karani, D., Introduction to Word Embedding and Word2Vec, in Towards Data Science. 2018.
8. Mikolov, T., et al. Efficient Estimation of Word Representations in Vector Space. 2013. arXiv:1301.3781.
9. Vaswani, A., et al. Attention Is All You Need. 2017. arXiv:1706.03762.
10. Devlin, J., et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 2018. arXiv:1810.04805.
11. Brown, T.B., et al. Language Models are Few-Shot Learners. 2020. arXiv:2005.14165.
12. Benaich, N. and I. Hogarth, State of AI Report 2020. 2020.
13. Metz, C., Meet GPT-3. It Has Learned to Code (and Blog and Argue). in New York Times. 2020.
14. Heaven, W.D., This avocado armchair could be the future of AI, in MIT Technology Review. 2021.
15. Radford at al., Connecting text and Images, in OpenAI blog. 2021.
16. Facebook AI's Multitask & Multimodal Unified Transformer: A Step Toward General-Purpose Intelligent Agents, IN Syc AI Technology and Industry Review. 2021.
17. Toews, R., 10 AI Predictions For 2021, in Forbes. 2020.
18. Fedus, W., B. Zoph, and N. Shazeer Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. 2021. arXiv:2101.03961.
19. Wiggers, K., Google trained a trillion-parameter AI language model, in Venture Beat. 2021.

For more information, please contact:

Dr. Svitlana Volkova
Chief Scientist, Decision Intelligence and Analytics
National Security Directorate
Pacific Northwest National Laboratory
svitlana.volkova@pnnl.gov

Dr. Tammie Borders
Technical Advisor, Data Science and AI
Defense Nuclear Nonproliferation R&D
Dept of Energy | NNSA
tammie.borders@nnsa.doe.gov

The DNN R&D AI and Data Science portfolio drives the development of next-generation AI methods and technologies to detect early indicators of nuclear weapons proliferation and reveal insights about the sophistication of existing nuclear weapons programs. Additionally, the Data Science portfolio develops technologies and practices to accelerate the use of AI-enabled technologies for national security missions across the U.S. government.