Clay Jones

Natural Language Processing (NLP) Techniques for Creating Artificial Intelligence (AI) Large Language Models (LLM)

Over the past few years, a lot of people have been finding out about LLMs, due to the growth of ChatGPT and Google Gemini. Even with the popular usage of these applications, many peoplearent aware of what an LLM even is. For this research project, I've been given the task of finding out what an LLM is, , how it is connected with A.I., and

First, LLM is short for Large Language Model. By definition, “A large language model (LLM) is a specialized type of artificial intelligence (AI) that has been trained on vast amounts of text to understand existing content and generate original content. [1]” How it works is the user gives the LLM a prompt and then the LLM returns an output based on patterns and language learned, similar to how generative A.I. ChatGPT and Google Gemini work. Based on a lecture from Google Cloud Tech, LLMs have many benefits such as having single models that can be used for many different tasks, fine-tuning processes requiring minimal field data, and performance growth with more data and parameters [2]. This gives LLMs many different use cases such as generating text, translating languages, organizing content, and chatbots [2]. One should also be cautious of LLMs because the data has a possibility of being biased and hallucinations, which is

when A.I interprets data and patterns that actually don't exist. Now that poses the question, how do LLMs learn and how are LLMs created? They learn and are created by the combination of A.I and NLPs.

Transitioning on, one of the biggest components of LLMs is Artificial Intelligence. To get a deeper understanding of LLMs, I had to learn about artificial intelligence, machine learning, and deep learning. These are all intertwined together. Deep learning is a subset of machine learning and machine learning is a branch of A.I. Based on a video by Machine Learning 101, “Artificial intelligence is the ability of a computer to perform tasks associated with humans, Machine Learning is A.I that focus on using data and algorithms to imitate the way that humans learn, and deep learning is a subset of machine learning methods based on neural networks. [4]” To go more in depth, while A.I. is focused on emulating human behavior like a chatbot for example, ML (Machine Learning) focuses on using data and algorithms to enable AI to imitate the way that humans learn, gradually improving its accuracy [3]. These algorithms mathematically represent real world processes. Use cases of machine learning are monitoring and customer support. Deep Learning is the most complex. DL (Deep Learning) is ML at the largest scale and tries to emulate the human brain. DL uses multiple algorithms to generate the best output through the use of neural networks. Neural networks are a complex topic but by definition, “it's a collection of algorithms designed to process data and produce an output with the least amount of error. [4]” Neural Networks have three layers, the input layers, the hidden layers, and the output layers. The input layers get a

prompt and the output layers give the result. The hidden layer determines the output based on the input based on the weighted error through forward propagation and backward propagation [4]. In simple terms, forward propagation is passing input data though to get a prediction and backward propagation adjusts the models parameters to minimize the loss by using optimized algorithms [4]. There are also many types of neural networks such as Feedforward neural networks, Convolutional neural networks, and Recurrent neural networks. A feedforward neural network (FNN) is a simple artificial neural network architecture in which data moves from input to output in a single direction [13]. FNN is the most simple neural work due to it not having feedback loops. The absence of feedback loops does not allow the FNN to remember already known data. This leads to less accuracy but faster results. A Convolutional Neural Network (CNN) is a specialized artificial neural network designed for image processing [13]. An example of a CNNs process is an image through grayscale. Each pixel is represented as a neuron and each pixel has a grayscale value. Afterwards, the image gets converted into a 1D vector for the computer to interpret. Recurrent Neural Network (RNN) is a type of Neural Network where the output from the previous step is fed as input to the current step [13]. RNN is the most effective and most recent neural network. This neural network focuses on the bidirectional signal flow with the use of feedback loops and uses long term memory to remember previous inputs. This helps come up with more accurate outputs.

Overall, A.I. is in charge of training an LLM. There are three different learning methods for A.I., unsupervised learning, supervised learning, and semi-supervised learning. These

learning algorithms analyze and group labeled and unlabeled data. Unsupervised learning discovers hidden patterns or data groupings without the need for human intervention [5]. Supervised learning does the same thing except with human intervention and labeled data. Semi-supervised learning occurs when only part of the given input data has been labeled [5]. The choice of these three algorithms decides how the data is cleansed, collected, and processed.

Furthermore, after covering A.I, I had to research the other component of LLMs, that begins with Natural Language Processing, or NLP for short. To research NLPs, I used an article by deeplearning.ai to give me a clear understanding. They determined that NLPs are the discipline of building machines that can manipulate human language — or data that resembles human language — in the way that it is written, spoken, and organized [6]. Based on natural language understanding and natural language generation, the overall goal of NLP is for a machine to gain an understanding and generate human language [6]. Some use cases of NLP provided by the article include information retrieval, text generation, autocomplete, and summarization [6]. With NLPs being so broad, the article includes many different applications of NLP. Two important ones highlighted are sentiment analysis, which is detecting emotion in data, and classification tasks, predicting the category or class of a given input based on training data [6].

Continuing, the deeplearning.AI model gave a lot of information about how NLPs work and NLPs work through a series of algorithms. There are specific steps that need to be

followed to make NLPs work. This process involves data preparation, data cleansing, and representing text numerically through vectors. The first step is to prepare the data. This involves gathering data and making a corpus. A corpus is a collection of texts or a dataset. This will be the data the NLP model will work with. The next step is data cleansing. This includes techniques like tokenization, stop word removal, lemmatization, punctuation removal and casing. These smaller units need to be interpreted as vectors so that the computer understands it [6]. Some of these units have little meaning and aren't useful for the computer to interpret, these are called stop words. Stop words include words like “the”, “a”, and “an” [6]. The machine needs to be able to detect base word forms like for example, “university,” “universities,” and “university’s” might all be mapped to the base univers [6]. Lemmatization is a more formal way to find roots by analyzing a word’s morphology using vocabulary from a dictionary [6]. The data also needs to be cleansed of punctuation and casing so that it's easier for the machine to interpret words in the corpus. These can be done through algorithms such as NLTK and spaCy, which both perform cleansing punctuation and casing. Through all this cleansing and preprocessing, this forms the vocab, which is our corpus after the preprocessing and cleansing.

NLP needs to represent text numerically through the use of vectors. A vector in short is a numerical representation of words that capture its meaning [9]. Through the use of vectors, the machine can embed words through web embedding. Word Embedding is a language modeling technique for mapping words to vectors of real numbers [7]. In all,

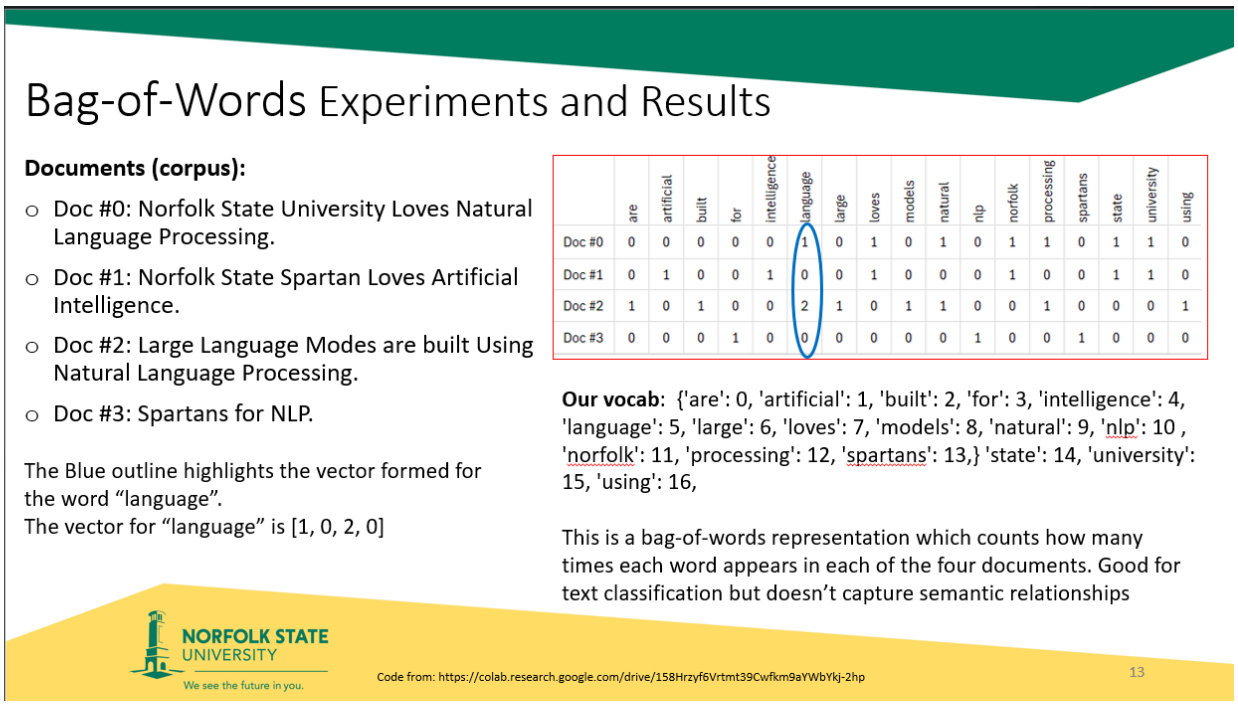
word embeddings help understand the context behind words. Before the words are embedded, the vocab has to find the importance and the weight of the tokenized data. In order to find the relevance of words during the NLP process, the bag-of-words and TF-IDF algorithms are used [6]. Bag-of-words counts the number of times a word is used in a document. It's simple and effective but doesn't capture the full relationship between words. However, Bag of words needed to be improved upon since it only counted how many times a word shows up. The TF-IDF algorithm is measured by term frequency, which is number or word occurrence divided by number of words in the document, and inverse term frequency, which is number of documents in corpus divided by number of documents with that word [6]. The TF-IDF is calculated by multiplying the TF times the IDF, which gives the weight of each word. Now that the vocab is weighed and put in vector form through these algorithms, the semantic relationships need to be captured between the vector and words of the vector space though word embedding.

Word2Vec consists of models for generating word embedding [7]. This algorithm uses vanilla neural networking to learn high dimension word embedding. In other words, Word2Vec is an effort to map words to high-dimensional vectors to capture the semantic relationships between words, developed by researchers at Google [7]. Semantics refer to learning the meaning of words and finding the context behind words. The combination of all these algorithms is what makes NLPs work. For Word2vec to work at its best, it uses the two architectures Continuous Bag of Words and Skip grams to generate high-dimensional vectors. Both of these algorithms use simple feed forward neural

networks. CBOW is an algorithm that aims to predict a target word based on its context words within a fixed window size [8]. Consider the word apple in a sentence. CBOW: Predicts the word “apple” based on the context words, such as “I ate a juicy ____.” The model learns that “juicy” and “ate” are likely context words [8]. Skip-gram, on the other hand, reverses the CBOW approach. Instead of predicting the target word from its context words, Skip-gram predicts the context words from the target word [8]. Skip-gram: Predicts context words like “juicy” and “ate” based on the target word “apple.” The model learns that “juicy” and “ate” are associated with “apple” [8]. In an experiment we did, we used a skip grams algorithm from google colab to make words using a pre cleaned data set [11]. The program took a word and mapped it to similar words in the data set. For example, if I typed in the word “burger” the words “steak” with a score of .678 and “pizza” with a score of .627. These are mapped together due to being food related, but steak would have the higher score because it's the most accurate. An alternative algorithm from word2vec used for word embedding would be gloVe. GloVe works very similar to word2vec but it does so by using matrix factorization techniques rather than neural learning [6]. Another key component that helps with NLP tasks are transformers. Algorithms like BERT provide contextual embeddings and enhanced knowledge representation.

As for our results, after experimenting on bag-of-words, TF-IDF, and word2vec, we were able to see the strengths and weaknesses of each algorithm and how each of them improve upon each other. Bag of words was the first one we covered but doesn't capture

any relationship between words. TF-IDF improved upon BoW because it gave weights to different words to attempt to capture a contextual relationship. The downside is that it doesn't capture a semantic relationship between words. Word2vec then improved upon both TF-IDF and BoWs. IT uses neural networks to generate word embeddings. Word2vec gives us the semantics but not the context. This was later improved because of transformers, something we will cover with future research. For our graphics, we created charts based on TF-IDF and Bag-of-words. We also showed the results of a skip grams word2vec algorithm.



	are	artificial	built	for	intelligence	language	large	loves	models	natural	nlp	norfolk	processing	spartans	state	university	using
Doc #0	0	0	0	0	0	1	0	1	0	1	0	1	1	0	1	1	0
Doc #1	0	1	0	0	1	0	0	1	0	0	0	1	0	0	1	1	0
Doc #2	1	0	1	0	0	2	1	0	1	1	0	0	1	0	0	0	1
Doc #3	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0

[10]

TFIDF Experiments and Results

$$TF(i,j) = \frac{\text{Term i frequency in document j}}{\text{Total words in document j}}$$

$$IDF(i) = \log_2\left(\frac{\text{Total documents}}{\text{documents with term i}}\right)$$

Documents (corpus):

- Doc #0: Norfolk State University Loves Natural Language Processing.
- Doc #1: Norfolk State Spartan Loves Artificial Intelligence.
- Doc #2: Large Language Modes are built Using Natural Language Processing.
- Doc #3: Spartans for NLP.

The TF-IDF vector for “language” is [0.363, 0, 0.534, 0]
The BoW vector for “language” is [1, 0, 2, 0]

Vocab	are	artificial	built	for	intelligent	language	large	loves	models	natural	nlp	norfolk	processing	spartan	spartans	state	university	using
TFIDF representation																		
doc0	0	0	0	0	0	0.363	0	0.363	0	0.363	0	0.363	0.363	0	0	0.363	0.46	0
doc1	0	0.453	0	0	0.453	0	0	0.357	0	0	0	0.357	0	0.453	0	0.357	0	0
doc2	0.338	0	0.338	0	0	0.534	0.338	0	0.338	0.267	0	0	0.267	0	0	0	0	0.338
doc3	0	0	0	0.577	0	0	0	0	0	0	0.577	0	0	0	0.577	0	0	0

- Assigns weight to words that hold more meaning
- Used to evaluate the importance of a word which is an improvement from BoW
- Limitation - Does not consider semantic meaning

NORFOLK STATE UNIVERSITY
We see the future in you.

Code from: https://colab.research.google.com/drive/1uy9y3l-bqTebD15Sq_jk3ayT5fRqNOYc

[11]

Word2vec Experimentation and Results

We used a pretrained Word2vec Skip Gram model trained using Google New’s 3-billion-word corpus dataset
We input the target words “Norfolk”, “College”, and “Burgers”, word2vec gives a list of related context words.
The scale rated from 0 to 1. The higher the number the more similar

```
model.most_similar('College')
```

```
[('Collge', 0.6822724938392639),
 ('University', 0.6669971346855164),
 ('Col_lege', 0.6303771138191223),
 ('Univeristy', 0.6178331971168518),
 ('Community_College', 0.6173429489135742),
 ('Unviersity', 0.5917240977287292),
 ('Univer_sity', 0.5827249884605408),
 ('Univerity', 0.5738999843597412),
 ('Colege', 0.5729310512542725),
 ('Colllege', 0.5718283653259277)]
```

```
model.most_similar('Norfolk')
```

```
[('Suffolk', 0.6745657920837402),
 ('Dorset', 0.6166641116142273),
 ('Essex', 0.6098291277885437),
 ('Yarmouth', 0.6097761988639832),
 ('Great_Yarmouth', 0.6018956899642944),
 ('Lowestoft', 0.6018952131271362),
 ('Del._Algie_Howell', 0.5961890816688538),
 ('Cornwall', 0.5862817168235779),
 ('Chichester', 0.5855712890625),
 ('Lincolnshire', 0.5795595645904541)]
```

```
model.most_similar('Burgers')
```

```
[('burgers', 0.690610408782959),
 ('Steak', 0.6767711043357849),
 ('Sandwiches', 0.6707561016082764),
 ('Hamburgers', 0.6643069982528687),
 ('burger', 0.6508228182792664),
 ('Steaks', 0.6507498025894165),
 ('Grill', 0.6418343782424927),
 ('Roast_Beef', 0.6286738514900208),
 ('Pizza', 0.6273280382156372),
 ('Bar_BQ', 0.6234582662582397)]
```

NORFOLK STATE UNIVERSITY
We see the future in you.

Code from: https://colab.research.google.com/drive/1VBs2ulhxZu_eKNZqHXoOG7Wb1Nv-nbBf

[12]

Sources

LLM
[1] “Large language models,” *www.gartner.com*. https://www.gartner.com/en/information-technology/glossary/large-language-models-llm

[2] Google Cloud Tech, “Introduction to large language models,” *YouTube*. May 08, 2023. [Online]. Available: https://www.youtube.com/watch?v=zizonToFXDs

ML
[3] IBM, “What is machine learning?,” IBM.com. https://www.ibm.com/topics/machine-learning

ML,AI,DL (neural)
[4] Machine Learning 101, “What’s the difference between AI, machine learning, and deep learning?,” YouTube. Aug. 11, 2020. [Online]. Available: https://www.youtube.com/watch?v=J4Qsr93L1qs

[5] IBM, “What is Unsupervised learning?” IBM.com What Is Unsupervised Learning? | IBM

NLP
[6] DeepLearning.ai, “Natural Language Processing (NLP) [A complete guide],” DeepLearning.AI, Jan. 11, 2023. https://www.deeplearning.ai/resources/natural-language-processing/

[7] “Python | Word Embedding using Word2Vec,” *GeeksforGeeks*, May 18, 2018. https://www.geeksforgeeks.org/python-word-embedding-using-word2vec/

[8] A. Verma, “Understanding CBOW vs. Skip-gram in Word Embeddings,” *Medium*, Nov. 06, 2023. https://ai.plainenglish.io/understanding-cbow-vs-skip-gram-in-word-embeddings-2d2f679dd755?gi=9a4995edfeea (accessed Jun. 20, 2024).

[9] S. Jeske and MarketMuse, “What is Word Vectors - Word Vectors Definition from MarketMuse Blog,” *MarketMuse Blog*.

https://blog.marketmuse.com/glossary/word-vectors-definition/#:~:text=Word%20vectors%20are%20vectors%20of (accessed Jun. 20, 2024).

[10] Google Colab “Bag-of-words” *Google* Clays_BOW.ipynb - Colab (google.com)

[11] Google Colab “TF-IDF” *Google* Clay-Tutorial-TF-IDF.ipynb - Colab (google.com)

[12] Google Colab “Word2Vec” *Google* other_word2vec.ipynb - Colab (google.com)

[13] “What is a neural network?,” *GeeksforGeeks*, Jan 03, 2024.What is a neural network? - GeeksforGeeks