NORFOLK STATE UNIVERSITY

NORFOLK STATE UNIVERSITY

We see the future in you.

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

Clay Jones

Wendon Doswell

Advised by Dr. Tonya Fields

Presentation

NORFOLK STATE UNIVERSITY

We see the future in you.

Introduction

Natural Language Processing (NLP) is a branch of artificial intelligence (AI) that allows computers to understand, generate, and manipulate human language.

For this project, we were given the task to research NLP techniques used to train Artificial Intelligence Large Language Models such as ChatGPT (OpenAI) and Gemini (Google)

Our overall goal is to create a Large Language Model through continued research during Fall 2024 semester

Methodology

1. Explore the connection between NLP, AI, and LLMs by researching machine learning, LLM creation, and NLP algorithms.
2. Explore background basics of NLP
   - Data Preparation – Prepare Corpus
   - Data Cleansing – Tokenization, Stop Words
3. Explore Data Modeling Techniques
   - One-hot encoding - Bag of Words – TF-IDF – Word2vec – BERT*
4. Report experiment findings

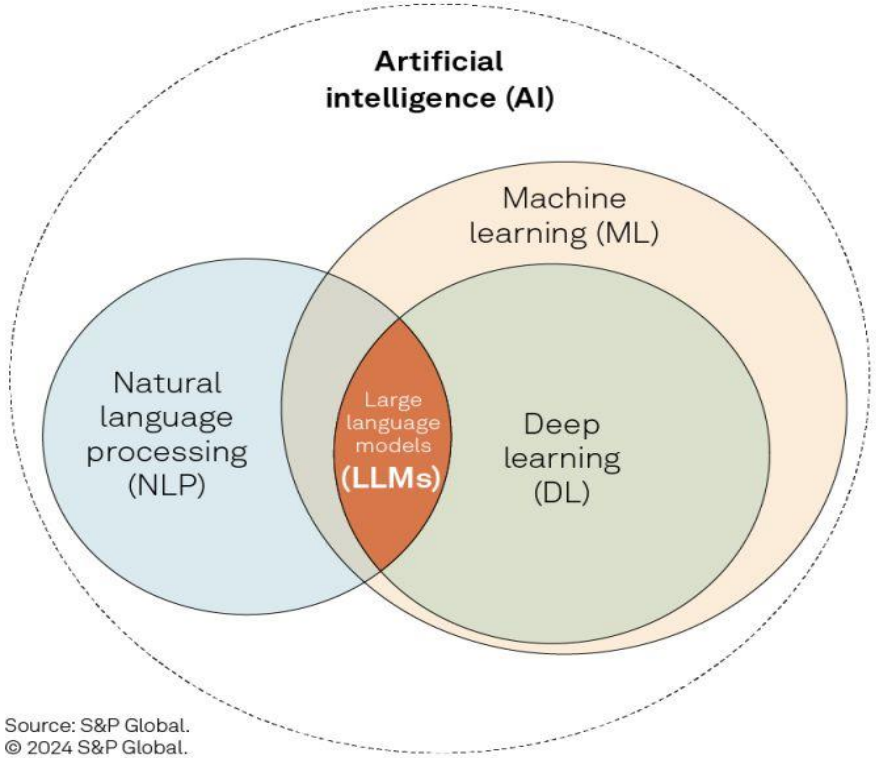
Libraries Used

- Scikit learn
  - A python library that implements machine learning models and statistical modelling.
  - Converts text into numerical representation using BoW
- TFIDF vectorizer class
  - Converts text into a matrix of tfidf
- Gensim library
  - This library provided us with a pretrained Word2vec model
- NLTK (Natural Language Toolkit) and spaCy
  - These algorithms get rid of spaces and cleanses the dataset of casing and punctuation.

Background - Connection between AI, NLP, and LLMs.

- Artificial Intelligence (AI) - is the science of creating intelligent machines capable of performing tasks typically performed by humans.
- NLP focuses on the interaction between computers and human language, allowing machines to understand and generate human language and responses. Science of representing text in a format processable by machine learning algorithms.
- Large Language Models (LLMs) is a specialized type of artificial intelligence (AI) that has been trained on vast amounts of text to understand existing contents used from various tasks and generate original content.

LLMs sit at the junction of NLP and deep learning



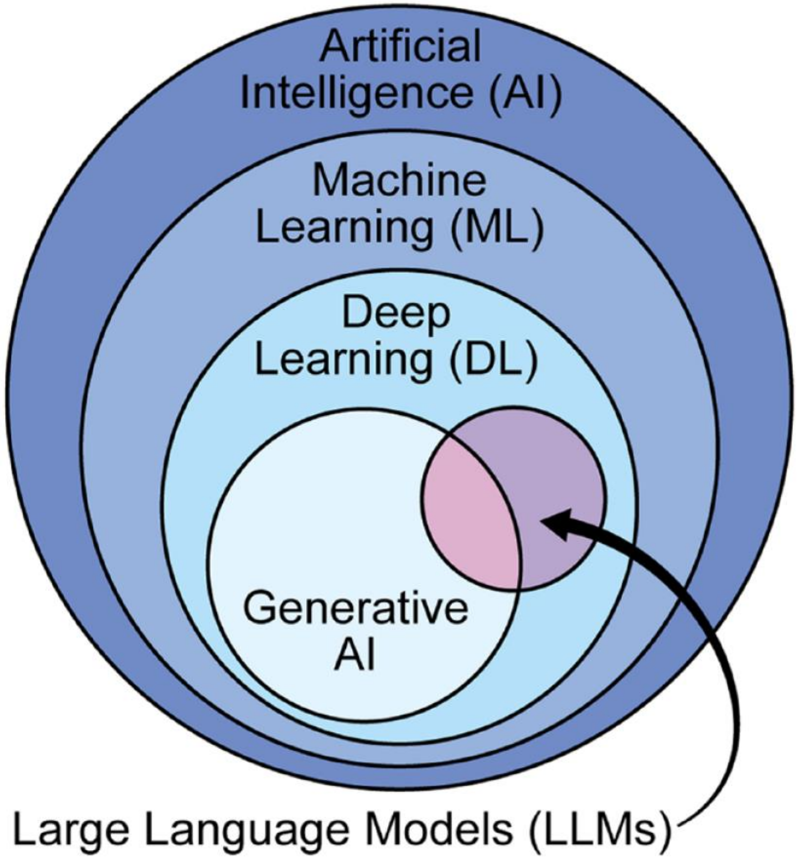
Source: S&P Global.

S&P Global

[Language Modeling: The Fundamentals | S&P Global (spglobal.com)]

Background - Connection between AI, NLP, and LLMs.

- Machine Learning
  - AI systems that automatically adapt and improve from experience without explicit programming. Uses statistical techniques to identify patterns. Makes decisions based on data
- Deep Learning
  - Subset of machine learning
  - Uses neural networks with multiple layers
  - Processes complex patterns in data
- Large Language Models
  - Natural Language Processing |natural language understanding|
  - Chatbot - chatgpt



From: https://www.researchgate.net/figure/LLMs-within-the-AI-taxonomy-LLMs-exist-as-a-subset-of-deep-learning-models-which-are-a_fig1_378394229

Background - Basics of NLP (Data Preparation)

- Create Corpus
  - A collection of documents to create a dataset
  - The corpus will be cleansed to create our vocab
- Tokenization
  - The process of breaking down text into smaller words or subwords called tokens.
    - “I like football” breaks down into ‘I’ , ‘like’, ‘football’.

- **Data Cleansing** - Involves removing irrelevant information in a dataset such as stop words and punctuation
- **Casing** - setting each word to have to same casing
- **Lemmatization** - Way to find root of words
- **Stop Words -** units or words that have little meaning and aren't useful for the computer to interpret
  - words like “the”, “a”, and “an”

Background - Data Modeling Techniques (Representing Text)

- Machines don’t understand raw text, so text needs to be represented numerically though vectors
  - A vector is a numerical representation of words [9]
- The text must be converted through algorithms such as
  - One hot encoding (Binary Encoded Vector)
  - Bag of words (Frequency Vector)
  - TFIDF (Frequency Vector)
  - Word2vec (Word Embeddings in Vector Space)
  - BERT* (Transformer)
  - GPT4* (Transformer)

NORFOLK STATE UNIVERSITY
We see the future in you.

Background - Data Modeling Techniques (Representing Text Algorithms)

Bag-of-Words

- To find the relevance of a word,
  - Bag-of-words counts the number of times a word is used in a document.
- Gives data a numerical representation to use for machine learning tasks
- Limitations
  - Doesn't capture full relationship between words
  - Doesn't factor in context

TF-IDF (Term Frequency-Inverse Document Frequency)

- Used to evaluate the importance of a word
- Improvements from BoW
  - Assigns weight to words that hold more meaning
- Limitations
  - Slow for large vocabularies
  - Does not consider semantic meaning

NORFOLK STATE UNIVERSITY
We see the future in you.

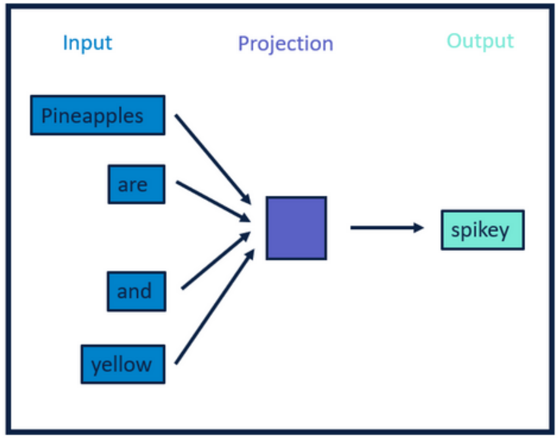
Background - Data Modeling Techniques (Representing Text Algorithms)

Word2vec

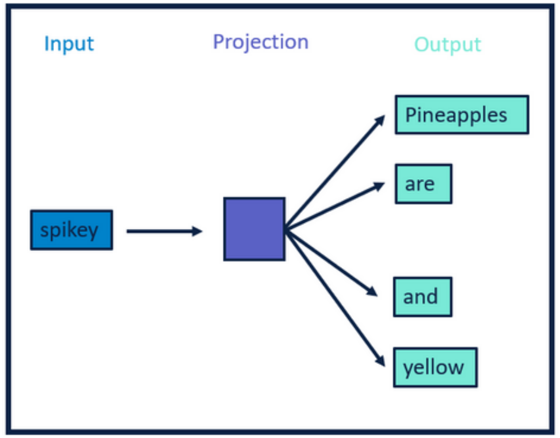
- Word2Vec is a pretrained model used for generating word embeddings [7]
- Maps words to high-dimensional vectors to capture the semantic relationships between words, developed by researchers at Google [7].
  - Added semantics which refer to learning the context behind words.
  - Word embeddings map words together in a vector space
- Word2vec makes use of the Algorithms
  - Continuous Bag of words (CBOW) and Skip-Grams

Word2vec

- CBOW is an algorithm that aims to predict a target word based on its context words
- Skip-grams, reverses the CBOW approach. Skip-grams predicts the context words from the target word [8].



CBOW



Skip-gram

Image Source: Word2VEC. Word2Vec Overview: | by Yusuf | Medium

Background - Data Modeling Techniques (Representing Text)

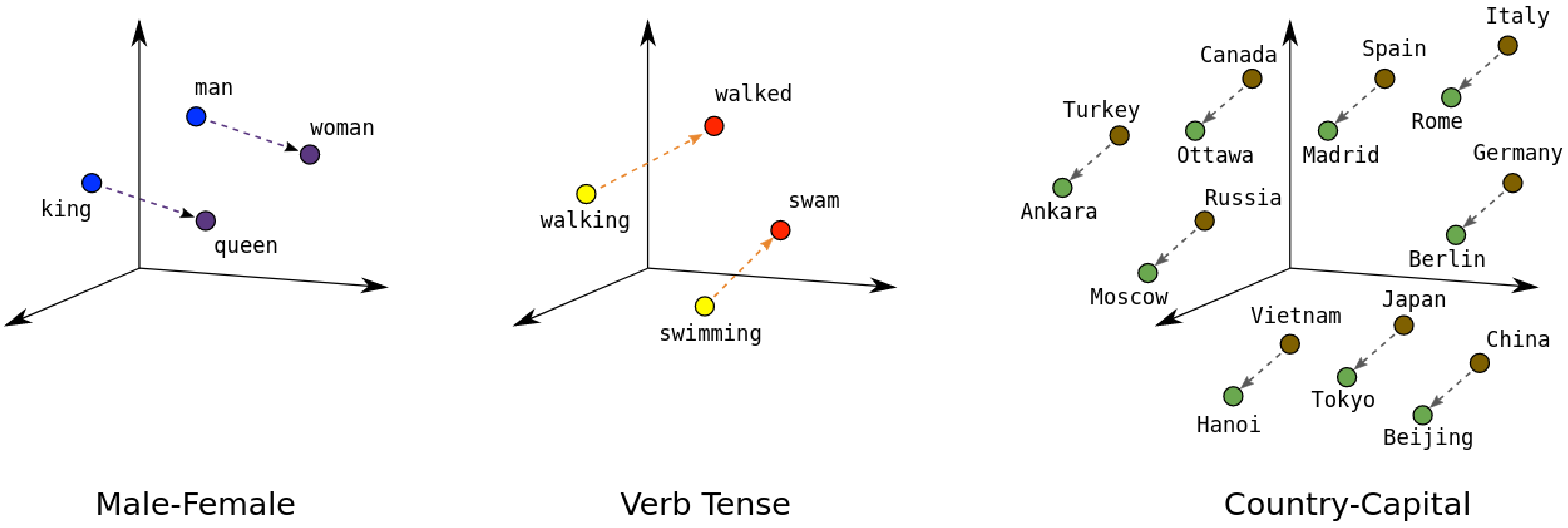


Image Source: [Word embeddings in 2020. Review with code examples | by Rostyslav Neskorozhenyi | Towards Data Science]()

Bag-of-Words Experiments and Results

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 Blue outline highlights the vector formed for the word “language”.
The vector for “language” is [1, 0, 2, 0]

	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

Our vocab: {'are': 0, 'artificial': 1, 'built': 2, 'for': 3, 'intelligence': 4, 'language': 5, 'large': 6, 'loves': 7, 'models': 8, 'natural': 9, 'nlp': 10 , 'norfolk': 11, 'processing': 12, 'spartans': 13,} 'state': 14, 'university': 15, 'using': 16,

This is a bag-of-words representation which counts how many times each word appears in each of the four documents. Good for text classification but doesn’t capture semantic relationships

Code from: https://colab.research.google.com/drive/158Hrzyf6Vrtmt39Cwfkm9aYWbYkj-2hp

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

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

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', 0.670756101608276 4),
 ('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

Conclusion

Research and Experimentation

- Data preparation and cleansing
- NLP text representation methodologies: (1) Bag-of-Words, (2) TF-IDF, and (3) Word2Vec

Findings

- 60 to 80 percent of training AI models is spent with data preparation and cleansing
- BoW - Doesn't capture full relationship between words
- TF-IDF - Evaluate the importance of a word but not the semantics
- Word2vec – gives semantics but no the context

NORFOLK STATE UNIVERSITY
We see the future in you.

Sources

[1] S. Arnold, "How to become an Accredited Investor on Linqto," *Private Equity Investing | Linqto Private Investing*, Mar. 27, 2024. What is Artificial Intelligence (AI) and Why it Matters (linqto.com)(accessed Jun. 05, 2024).

[2]"Bag of Words Model in NLP Explained," *Built In*, 2023. https://builtin.com/machine-learning/bag-of-words (accessed Jun. 05, 2024).

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

[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).

NORFOLK STATE UNIVERSITY

We see the future in you.

Questions?