
# Implementing AI in Developing an Ontology for Digital Thread Integration Solution

Nicole Manno
Digital Engineer | Intelligent Systems Engineering (ISE)

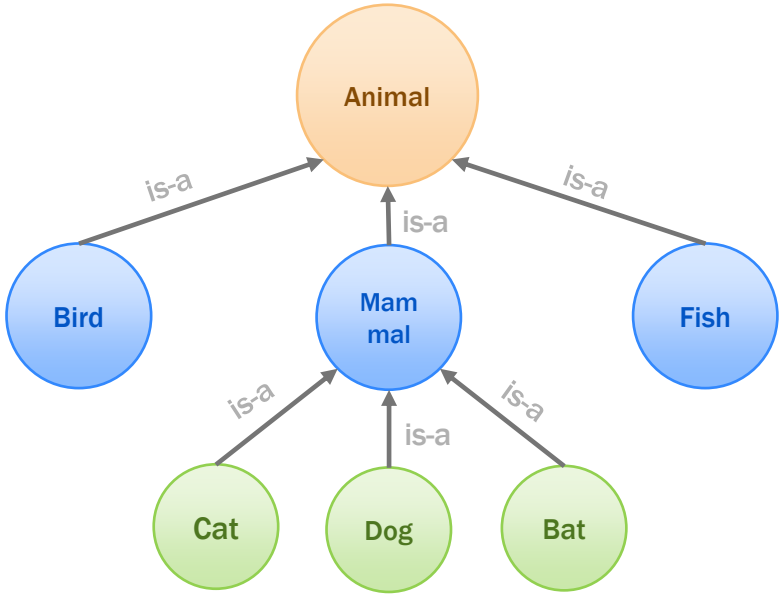
# Agenda

# Understanding the Digital Thread

- The **Digital Thread** is the concept of interconnecting data from every phase of a product's lifecycle via a data-driven architecture of shared resources to facilitate real time and long-term decision making
- Ontologies play a crucial role in structuring and making sense of the vast amount of data involved in the Digital Thread

# The Role of Ontologies

- In information science, **ontologies** are formal representations of knowledge as a set of concepts within a domain, and the relationships between those concepts
- Can be useful for data integration, information retrieval, and in reasoning about the domain



# The Challenges in Ontology Development

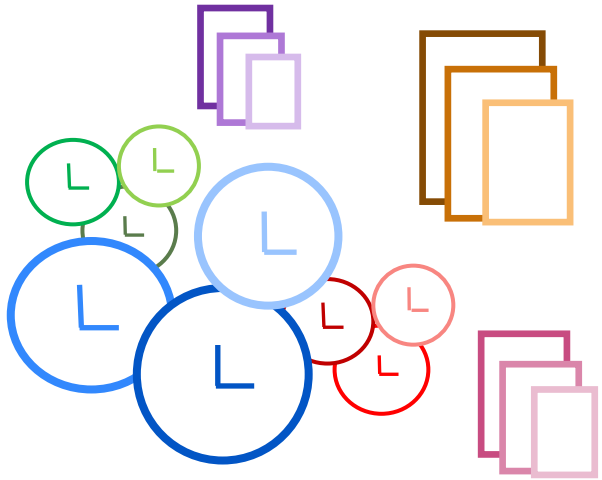
- **Challenges**
  - Domain Complexity
  - Maintaining Consistency
  - Ensuring Scalability
- **Artificial Intelligence (AI) and Large Language Models (LLM) can serve as the starting point in ontology development**
  - Can assist in identifying key concepts within a domain
  - ***Note:*** *that AI may not be the complete solution and human expertise is still required*

# Artificial Intelligence and Large Language Models

- **Artificial Intelligence** (**AI**) is the simulation of human intelligence processes by machines
- **Large Language Models** (**LLMs**) are a subset of AI, trained on a large amount of text data to "understand" and generate human-like text
- **Prominent LLMs in the Industry**
  - OpenAI's GPT-series
  - Google's Vertex AI and Bard

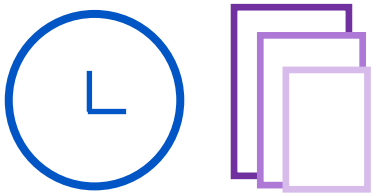
# Fine-Tuning LLM vs. Fine-Tuning from Scratch

From scratch: Developing and training a model from scratch for a specific task



VS.

Pre-trained: Using a pre-trained LLM and fine-tuning for a specific domain or for a specific task



**Fine-tuning a pre-trained model is faster and requires a lot less data compared to training a model from scratch**

# Bloomberg's Domain-Specific LLM

- On March 30, 2023, Bloomberg released the finance-domain LLM BloombergGPT
- The training corpus for the model contains over 700 billion tokens that was created from a public dataset containing 345 billion tokens

*"BloombergGPT outperforms similarly-sized open models on financial NLP tasks by significant margins – without sacrificing performance on general LLM benchmarks"*

# Key Objective

**Develop a taxonomy for a Digital Thread ontology by fine-tuning a Large Language Model**

## Project Scope & Tools

- The ontology built will be focused on systems engineering process rather than the system itself.
- Leveraging Google's Vertex AI models for the fine-tuning process.
- **Note:** This is an ongoing initiative. Our current focus is establishing a taxonomy. Addressing relationships and instances will be a future endeavor

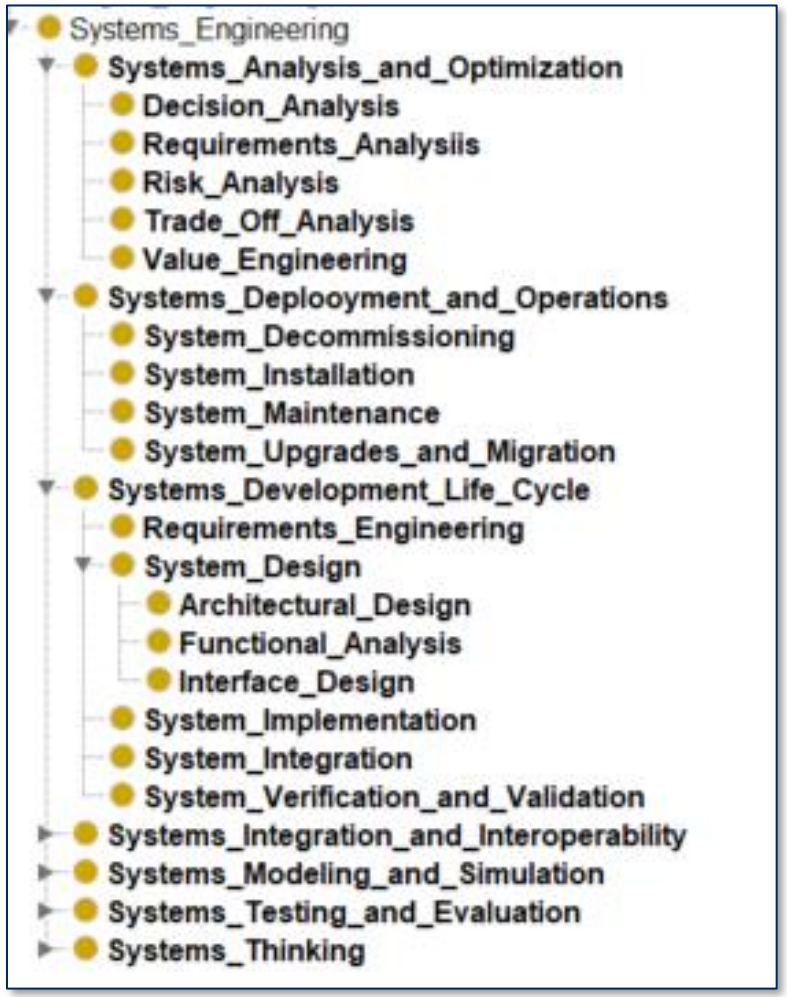
# Prompt

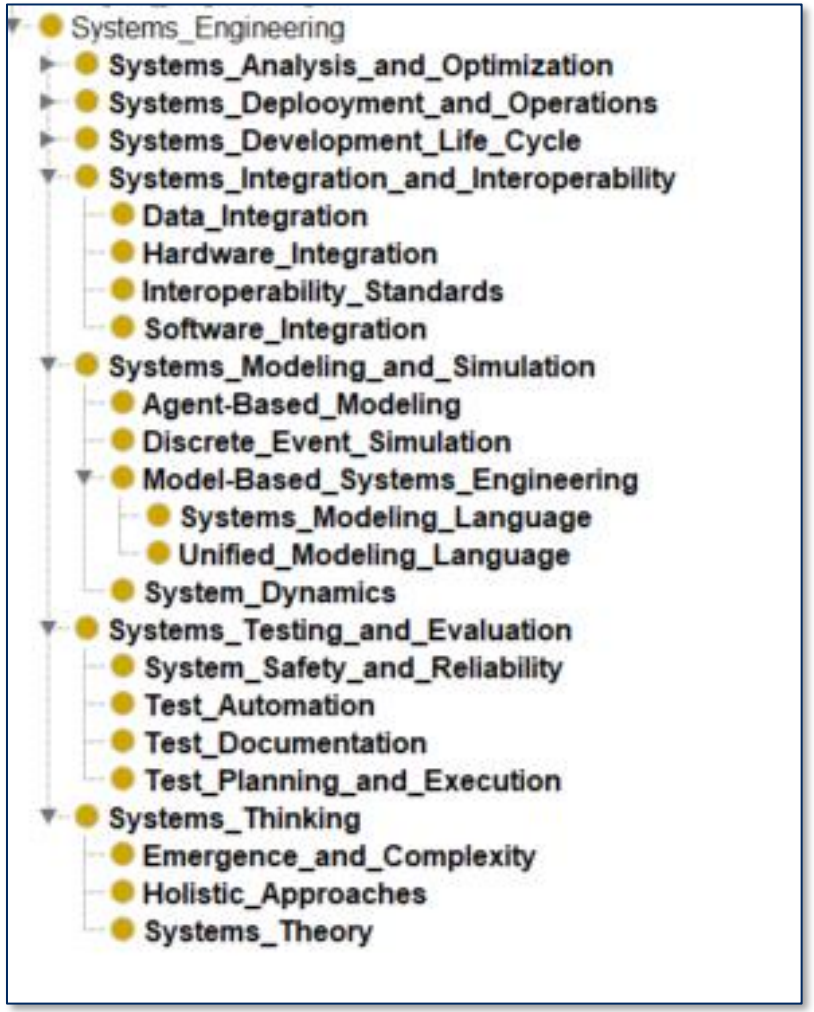
*“Please generate a comprehensive taxonomy for a Systems Engineering (SE) and Digital Engineering (DE) reference ontology. Begin with primary categories, and for each, detail multiple layers of subcategories. Ensure that the structure is deep, capturing the core concepts, methodologies, tools, and intricacies unique to both SE and DE, down to the most granular levels where possible.”*

# Metrics of GPT-3 Prompting

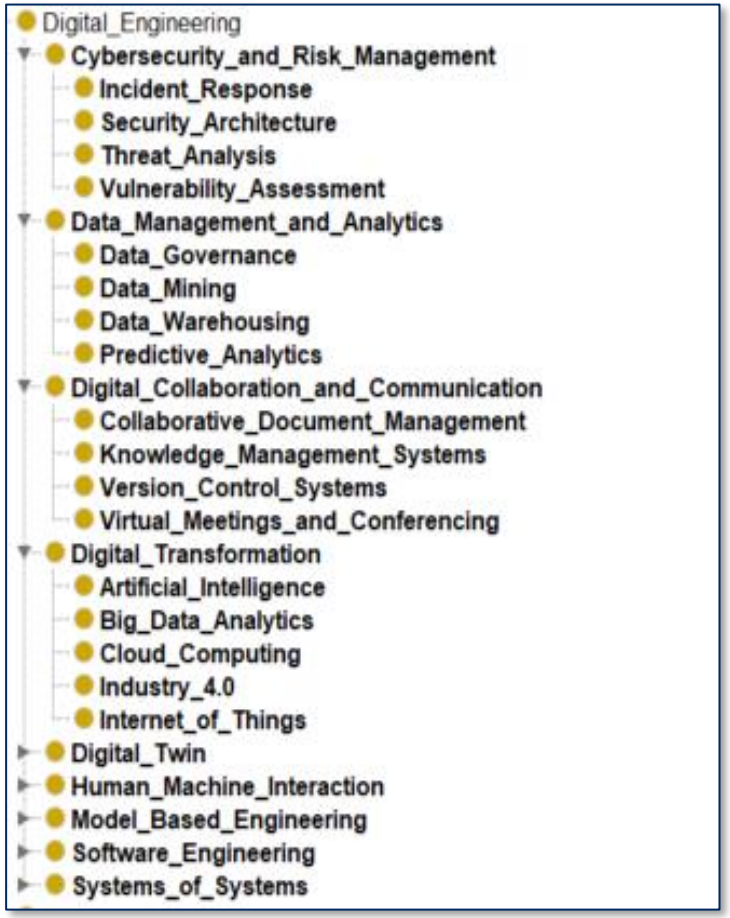
- **Model:** gpt-35-turbo
- **Parameters**
  - Temperature: 0.70
  - Token Limit: 4000
  - Top-P: 0.95

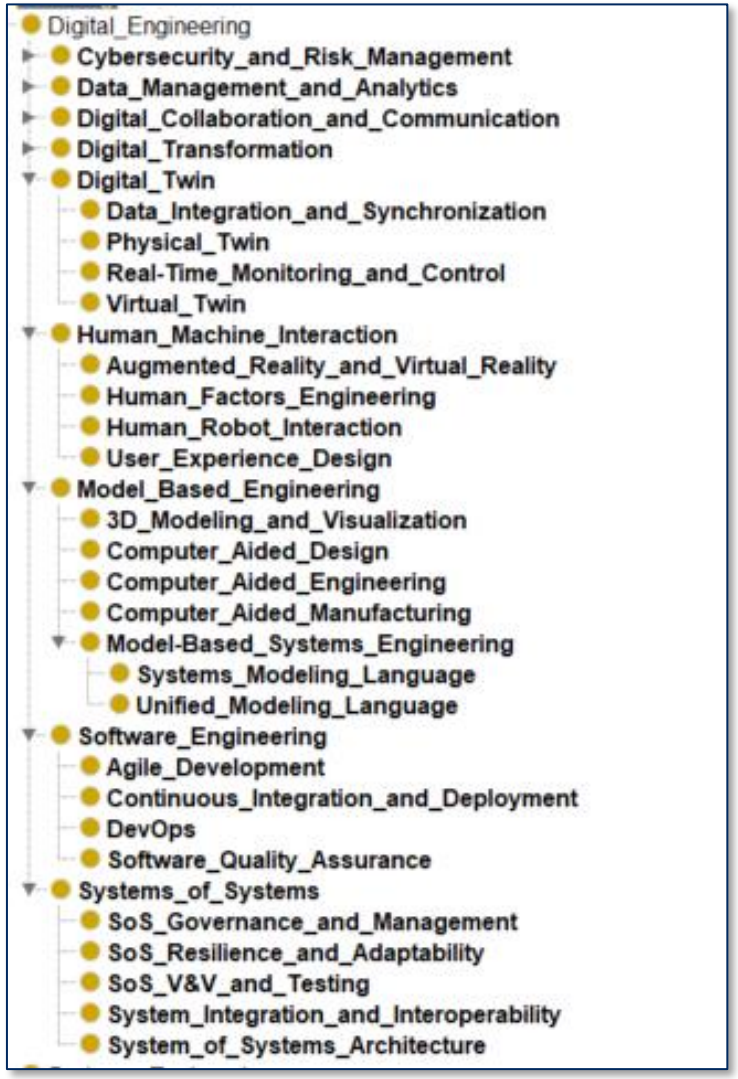
# GPT 3 - Zero-Shot Learning



# GPT 3 - Zero-Shot Learning (cont.)

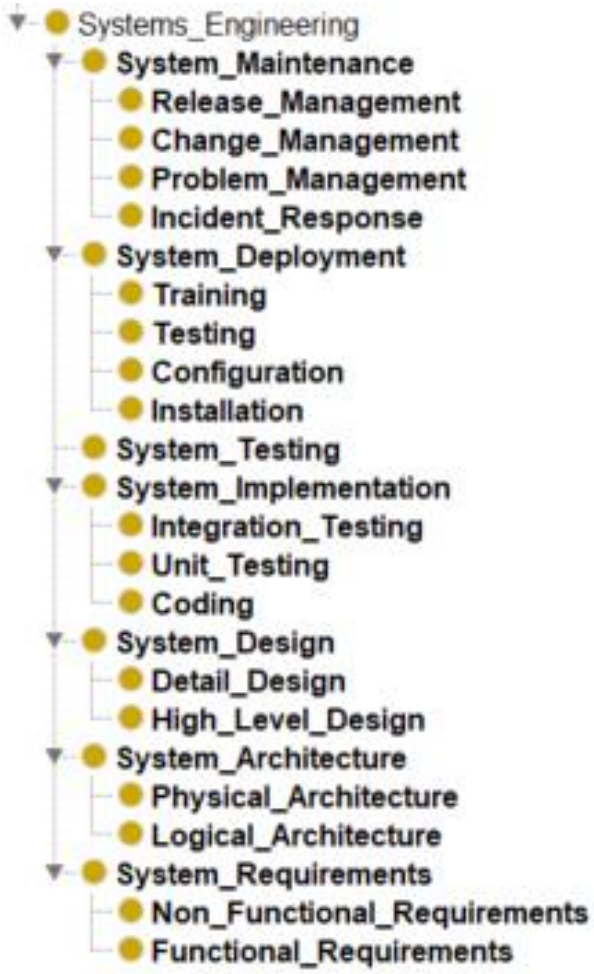


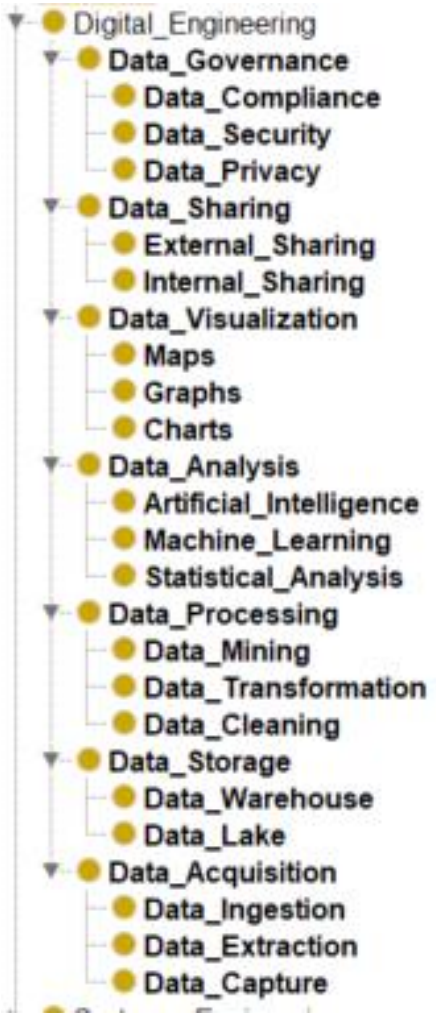



# Metrics of Vertex AI Prompting

- **Model**: chat-bison@001
- **Parameters**
  - Temperature: 0.20
  - Token Limit: 1024
  - Top-K: 40
  - Top-P: 0.80

# Vertex AI - Zero-Shot Learning



# Methodology for Fine-tuning

Collect SE Resources

Prepare the Data

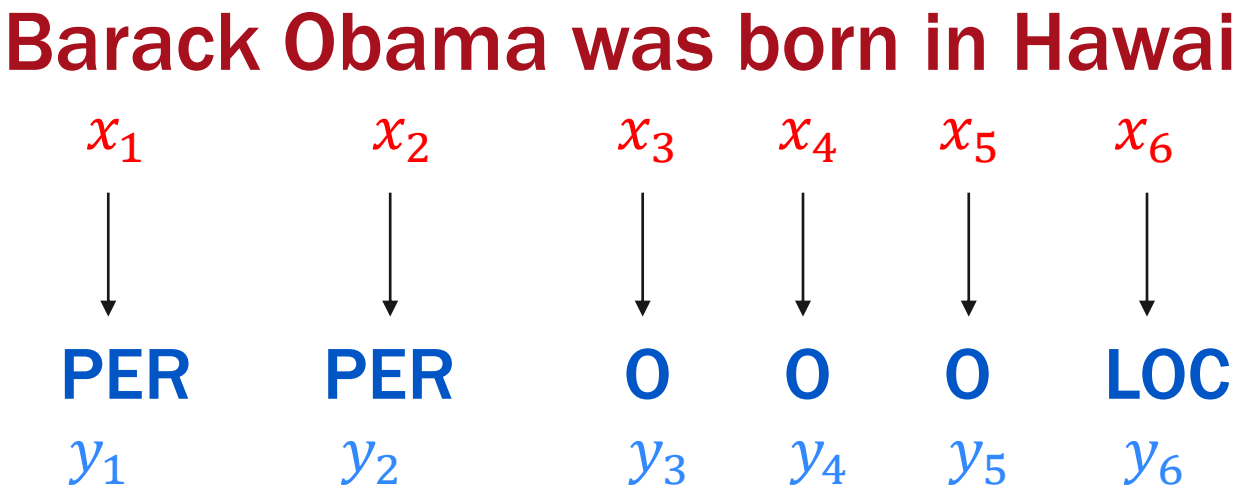
Fine-Tune the LLM

# Types of Machine Learning

- **Supervised**
  - Trained on labeled data (input & corresponding output)
- **Unsupervised**
  - Trained on data without explicit labels
- **Self-Supervised**
  - Model generates its own supervisory output from input
- **Semi-Supervised**
  - Uses both labeled and unlabeled data

# Named Entity Recognition (NER)

**Named Entity Recognition** (**NER**) is an information extraction task that identifies certain entities from a sentence/paragraph/document.



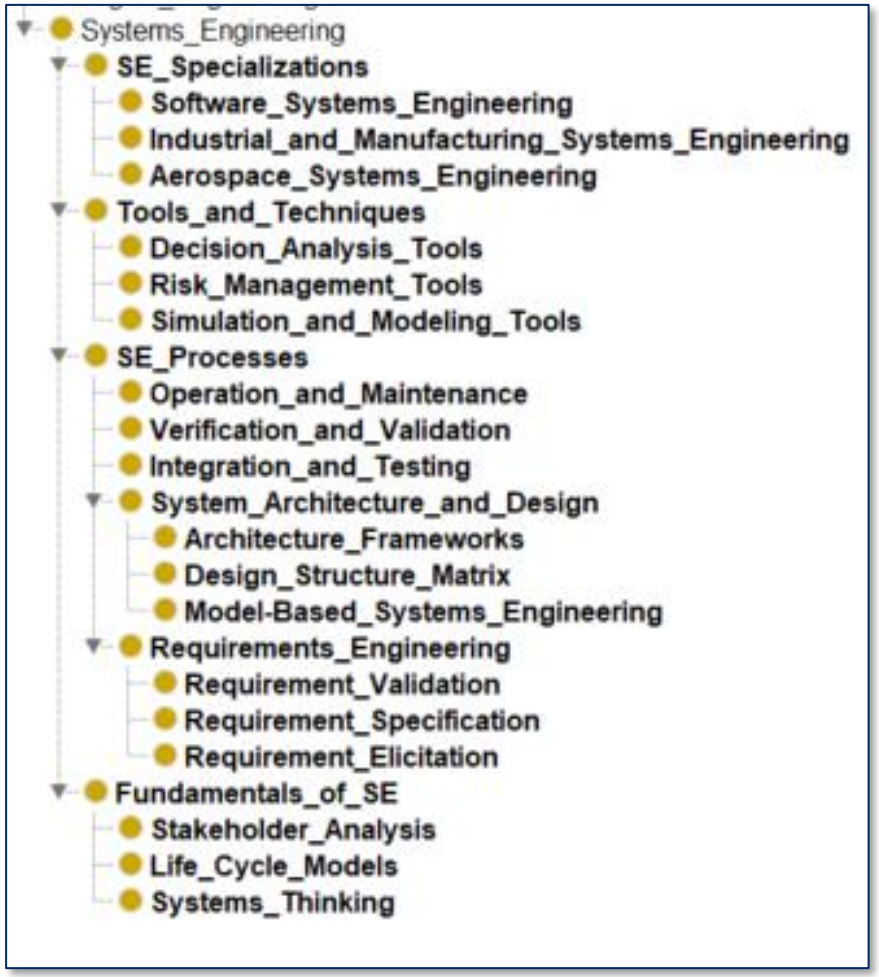
# Annotating Data

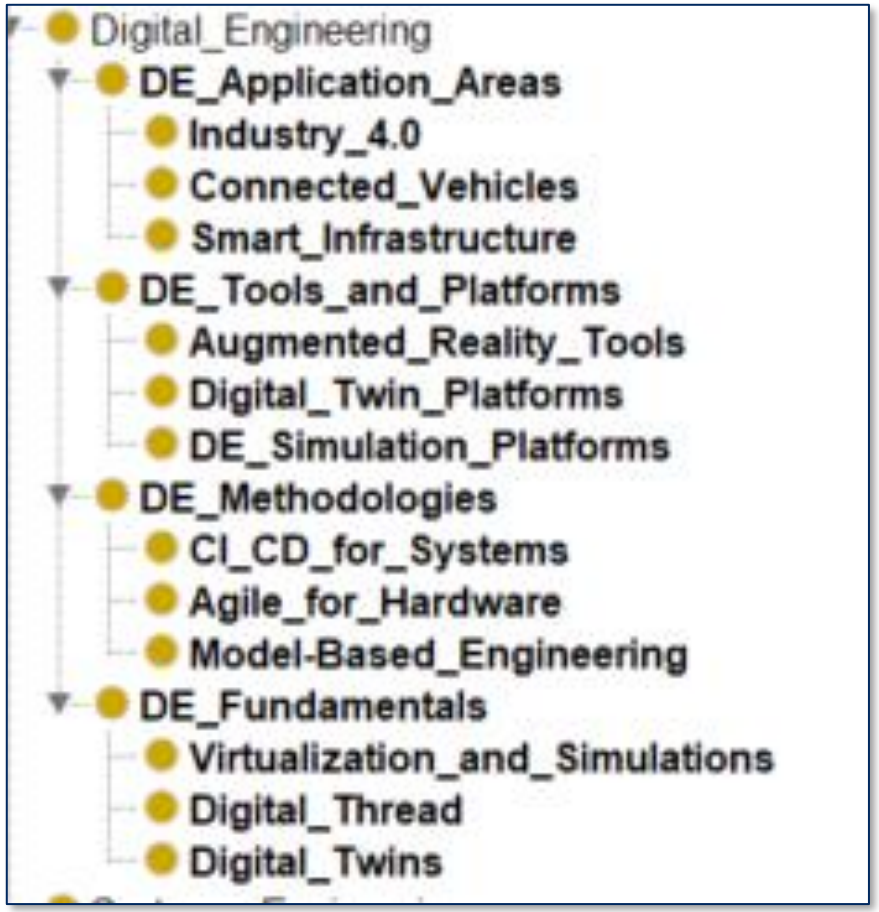
- **Resources**
  - Systems Engineering Book of Knowledge (SEBoK)
  - INCOSE SE Handbook
  - ISO/IEC/IEEE 15288:2015
  - ManTech Proprietary Data
- **Annotations**

# Metrics of Supervised Results

- **Time:** 62.37 minutes
- **Data**: 200
- **Model:** text-bison@001
- **Parameters**
  - Temperature: 0.2
  - Token Limit: 1024
  - Top-K: 40
  - Top-P: 0.80

# Supervised Machine Learning Results



# Supervised Machine Learning Results (cont.)

# Next-Token Prediction

- **Next-Token Prediction** is a self-supervised ML method that involves predicting the most likely subsequent word or token in a sequence based on the preceding context
- LLMs like GPT-series leverage this technique to generate coherent and contextually relevant text

$$S = \{s_1, s_2, \ldots, s_n\}$$

Sequence of tokens

$$\mathcal{L}(S) = \sum_{i=1}^{n} \log(\wp(s_i | s_{i-k}, \ldots, s_{i-1}, \theta))$$

The probability of $s_i$ given $k$ preceding tokens and model parameters $\theta$

# Next-Token Prediction (cont.)

- Idea: given a sequence of tokens, the LLM predicts the next token.
- **Example**

"Barack Obama was born in Hawaii"

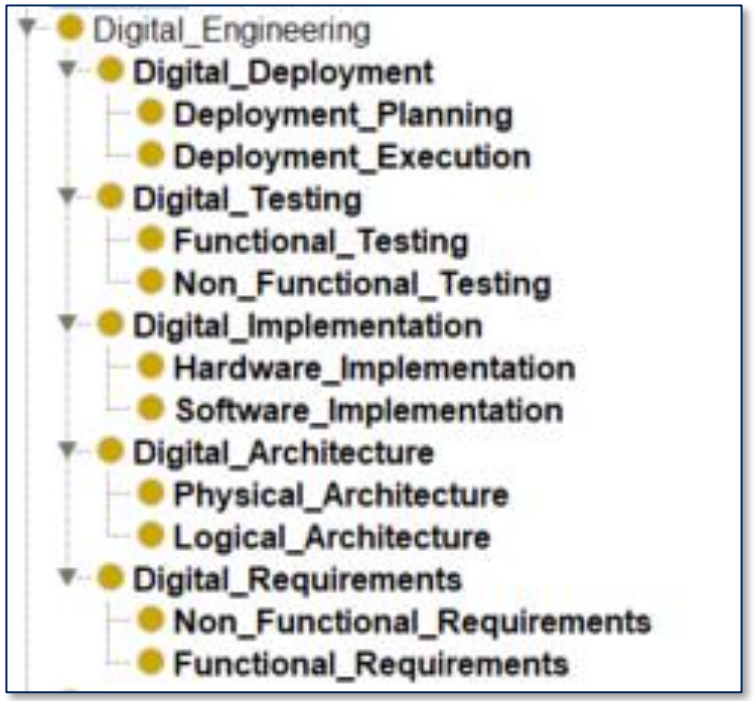
["Barack", "Obama", "was", "born", "in", "Hawaii"]

| Prompt | Completion |
| --- | --- |
| Barack | Barack Obama |
| Barack Obama | Barack Obama was |
| Barack Obama was | Barack Obama was born |
| Barack Obama was born | Barack Obama was born in |
| Barack Obama was born in | Barack Obama was born in Hawaii |

# Metrics of Self-Supervised Results

- **Time:** 62.28 minutes
- **Data:** 3405
- **Model:** text-bison@001
- **Parameters**
  - Temperature: 0.20
  - Token Limit: 1024
  - Top-K: 40
  - Top-P: 0.80

# Self-Supervised Machine Learning Results

# Observations

- **Performance**
  - Supervised Fine-Tuned Model performed the best, with a consistent and comprehensive breakdown
  - Fine-tuned next token prediction was notably more generalized
- **Lack of Hierarchical Depth**
  - No taxonomy delve deep into multiple levels of subcategories, could be attributed to prompting
- **Model hallucinations**
  - Vertex AI Fine-tuned with Next-Token Prediction experienced hallucinations

# Lessons Learned

- **Importance of Effective Prompting**
  - Crafting the right prompts is vital for meaningful and precise responses
- **Quality of Fine-Tuning Data Matters**
  - The efficacy of a fine-tuned model is linked to the quality and relevance of training dataset
- **Acknowledging Bias in Annotation**
  - Recognizing and addressing potential biases from human and AI annotations
- **Web and Document Scraping Considerations**
  - Exercise caution when scraping as resources can prohibit it

# The Future of AI

- **Ubiquity of AI in Daily Living**: AI will become an integral part of our daily lives
- **Continuous Evolution and Expansion**: AI capabilities will continually expand and enhance with ongoing research
- **AI's Role in Systems Engineering**: AI will be increasingly tailored to address systems engineering business challenges

# References


- Bloomberg. (2023, March 30). "Introducing BloombergGPT, Bloomberg's 50-billion parameter large language model, purpose-built from scratch for Finance." Bloomberg LP Press. [Online]. Available: https://www.bloomberg.com/company/press/bloomberggpt-50-billion-parameter-llm-tuned-finance/.
- Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... & Agarwal, S. (2020). "Language Models Are Few-Shot Learners." OpenAI. [Online]. Available: https://www.openai.com/research-publications/
- INCOSE. (2020). "Systems Engineering Book of Knowledge (SEBoK)," version 2.4. International Council on Systems Engineering.
- INCOSE. "Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities," version 4.0. International Council on Systems Engineering.
- ISO/IEC/IEEE 15288. (2015). "Systems and Software Engineering – Life Cycle Processes." Institute of Electrical and Electronics Engineers Standards. [Online]. Available: https://standards.ieee.org/standard/15288-2015.html.
- Jaynes, E. T. (2003). "Probability Theory: The Logic of Science." Cambridge University Press.
- MacKay, D. J. C. (2003). "Information Theory, Inference, and Learning Algorithms." Cambridge University Press.
- Orellana, D. & Mandrick, W. (2019). "The Ontology of Systems Engineering: Towards a Computational Digital Engineering Semantic Framework." Procedia Computer Science, Volume 153.
- Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). "Improving Language Understanding by Generative Pretraining." OpenAI. [Online]. Available: https://www.openai.com/research-publications/
- Singh, V. & Willcox, K. E. (2018). "Engineering Design with Digital Thread." 2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 8-12.

# For more information contact:

Dr. Douglas Orellana, Douglas.Orellana@ManTech.com

Nicole Manno, Nicole.Manno@ManTech.com