LLM-Augmented Pipelines for Validated System Model Creation

Extracting and Structuring Complex System Representations from Text

Carlo Lipizzi

2025

Context

- There is an unprecedented complexity in mission systems that demand new digital engineering approaches
- Model-Based Systems Engineering (MBSE) and Digital Engineering (DE) are at the core of this transformation, supported increasingly by Artificial Intelligence
- Large Language Models (LLMs) present powerful opportunities but introduce new challenges for ensuring trustworthiness in defense environments
- This presentation outlines pipelines for transforming unstructured data into structured models suitable for operations

Core Objective

- The primary objective is to create semi-automated pipelines that convert unstructured documentation and/or traditional SE models into structured, verifiable and interactive systems
- This is achieved by combining the scalability of LLM-based extraction with subject matter expert validation and advanced graph-based analysis
- The outcome includes validated dependency graphs, mission models and knowledge bases that can be deployed in operational workflows. These results directly support programs by enabling AI augmentation of MBSE and DE in a trustworthy manner

Research Foundation

- This work builds on prior research presented in 'From Text to Structure' (Lipizzi, 2025), which established feasibility for extracting structured knowledge from text
- The present effort extends those methods to not only generating models but also to embedding assurance through validation and feedback mechanisms
- These elements are critical for adoption in contexts such as the Defense, where transparency and explainability are required

Methodology Overview

- The methodology includes five key steps:
  - data collection
  - knowledge extraction using LLMs
  - semantic graph construction
  - SME validation
  - embedding for analysis
- This pipeline ensures that every stage includes mechanisms to identify and correct errors before they propagate further
- Recent extensions also include integration with Graph Neural Networks (GNNs) to allow deeper structural reasoning and feedback on requirements
- Together, these elements provide both scalability and assurance for relevant digital engineering workflows

Step 1: Data Collection

- The pipeline begins with systematic data collection from heterogeneous sources such as specifications, manuals, SysML models and legacy reports
- Three different types of inputs have been tested: natural language requirements written in plain English, data extracted from a system modeling tool and UML/XML/SysML files representing systems
- This diversity ensures that the process works across the range of artifacts currently available in different engineering practices

Step 2 & 3: LLM Extraction and Graph Construction

- In this step, Large Language Models extract subject–predicate–object triplets that represent key entities and relationships within the system
- These outputs are then transformed into semantic/knowledge graphs that visualize dependencies, interactions and structures
- When interacting directly or indirectly with a system modeling tool, the extraction of nodes/edges is performed directly, without the use of LLMs
- The graph representations enable engineers to explore system knowledge more systematically and intuitively, identifying possible gaps or contradictions
- The structured graphs form the foundation for subsequent expert validation and machine reasoning using GNNs

Step 4: SME Validation

- The extracted graphs are presented to Subject Matter Experts through graphical interfaces for validation and correction. Gephi has been used in the prototypes
- SMEs ensure that the model aligns with mission-critical truths, technical accuracy and operational requirements
- This process establishes a trusted baseline that combines automation with expert oversight, increasing both efficiency and assurance
- Validated models then serve as reliable inputs for deeper analysis, embedding and downstream reasoning tasks

Step 5: Embedding, GNN Analysis, and Querying

- Validated graphs are embedded into vector spaces that enable semantic search and retrieval-augmented generation (RAG), grounding AI in trusted data
- At this stage, Graph Neural Networks (GNNs) are introduced to perform advanced reasoning tasks such as classification, link prediction and semantic inference
- Key advantages of GNNs include the ability to detect missing or incorrect relationships, refine embeddings and learn contextual representations of requirements
- Most importantly, GNNs provide structured feedback that can be used to evaluate, refine and improve the original requirements themselves

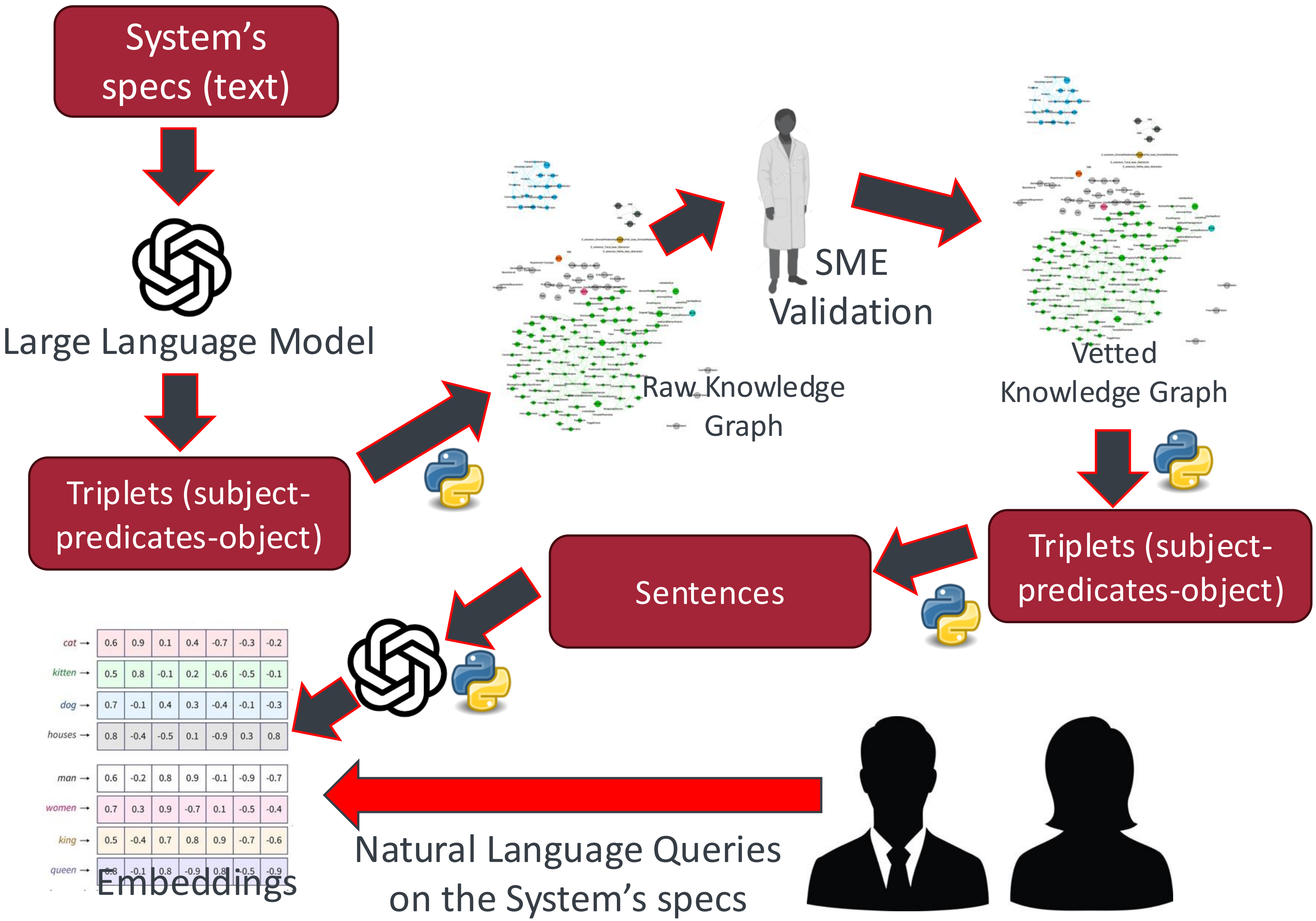
Side benefit: Natural Language requirements queries

- The system can also be used to create plain English queries to the requirements
- The knowledge graph representing the system requirements can be transformed into sentences. Sentences can be vectorized, creating a computational knowledge base for the system, that can be queried by matching user's queries with vectors in the knowledge base
- This approach can also be used to validate for trustworthiness/quality assurances the content of the knowledge base
- This step has been successfully prototyped

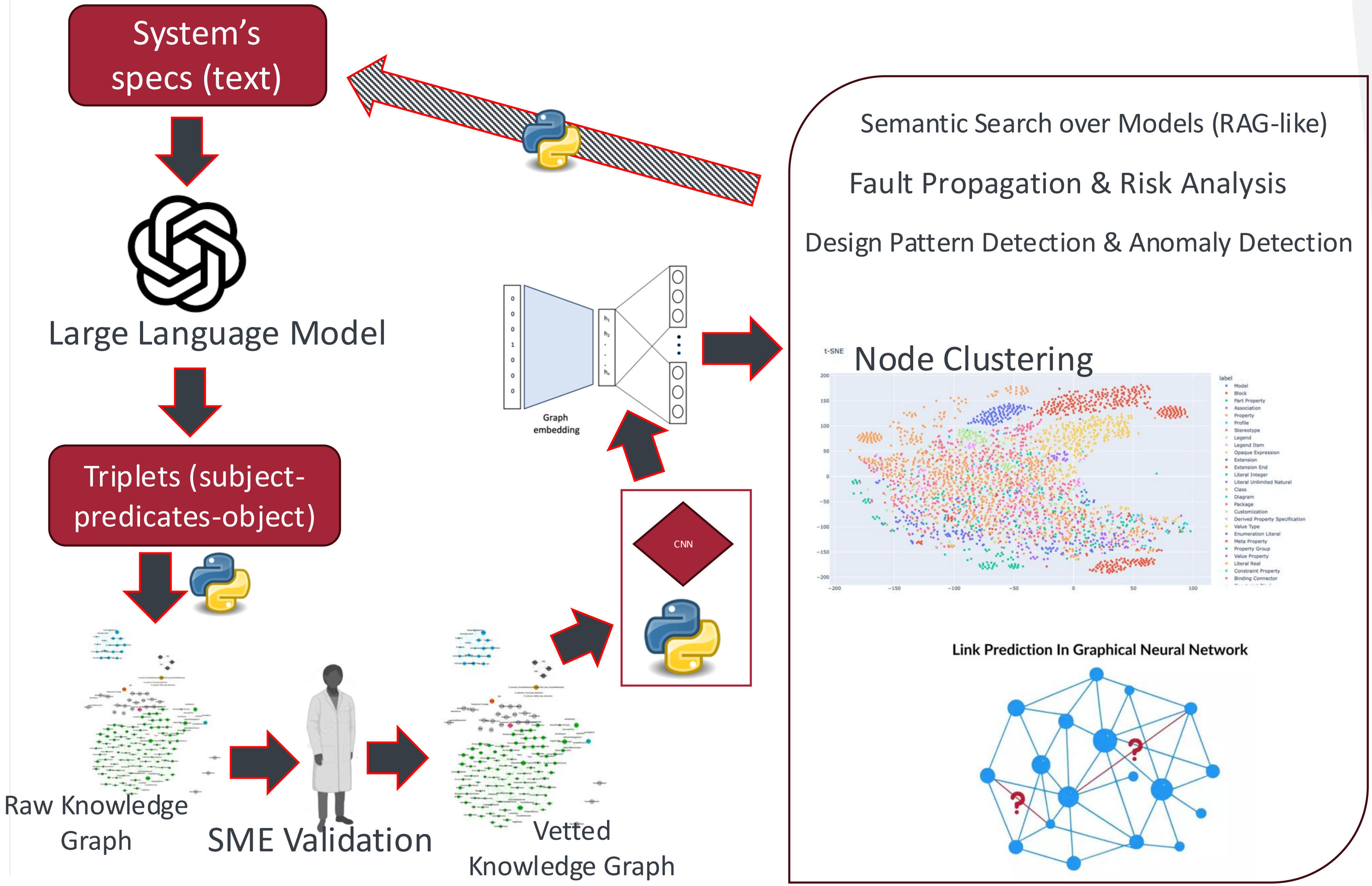
Prototypes Case Study

- The prototypes pipelines have been tested to demonstrate feasibility and practical impact
- LLM-extracted graphs were validated and then enhanced using GNNs, which identified potential inconsistencies in relationships
- The current prototypes does not close the loop, missing the recreation/revision of the requirements

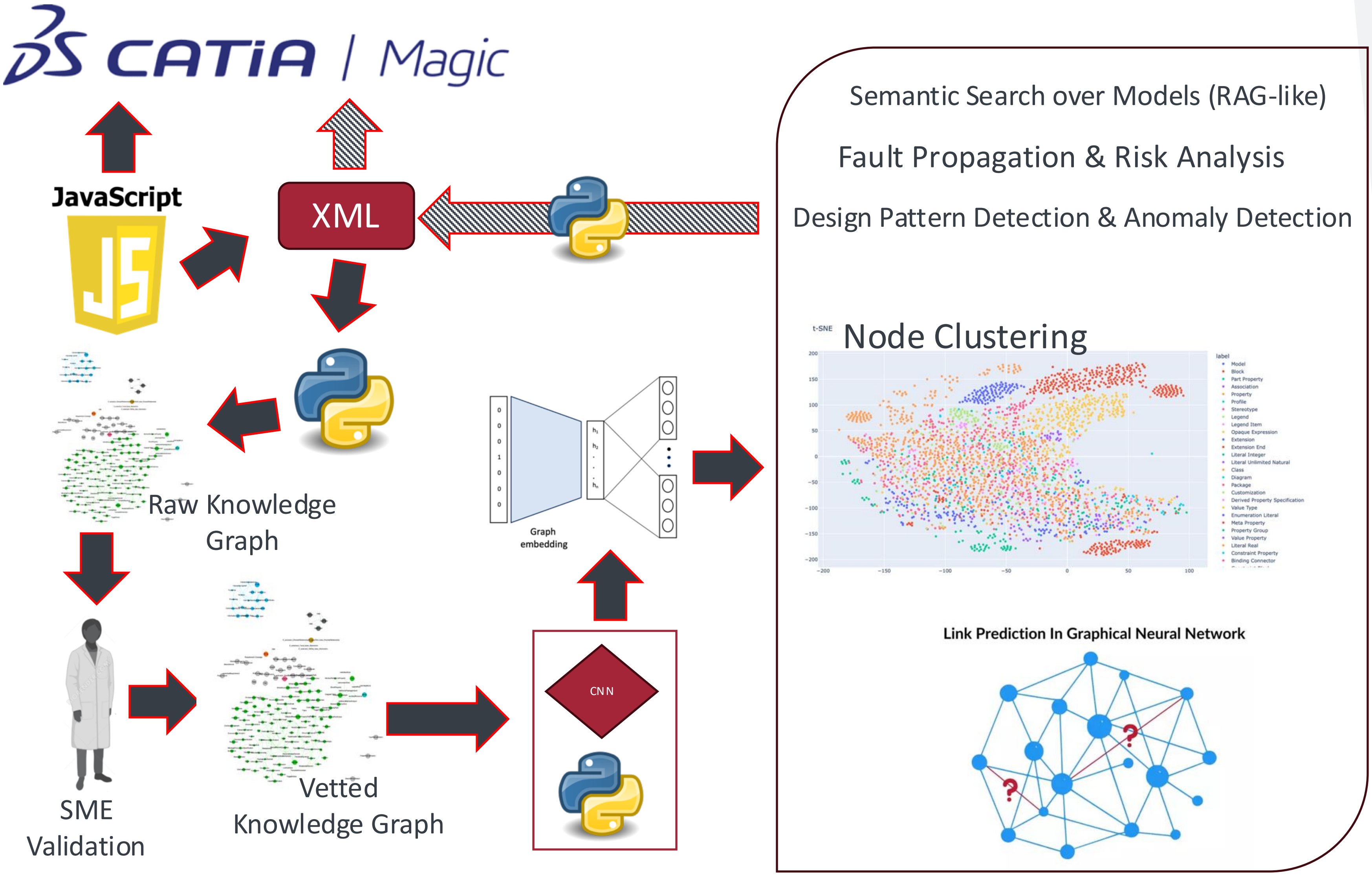
Prototype Case Study 1 – Using System's Specs - 1



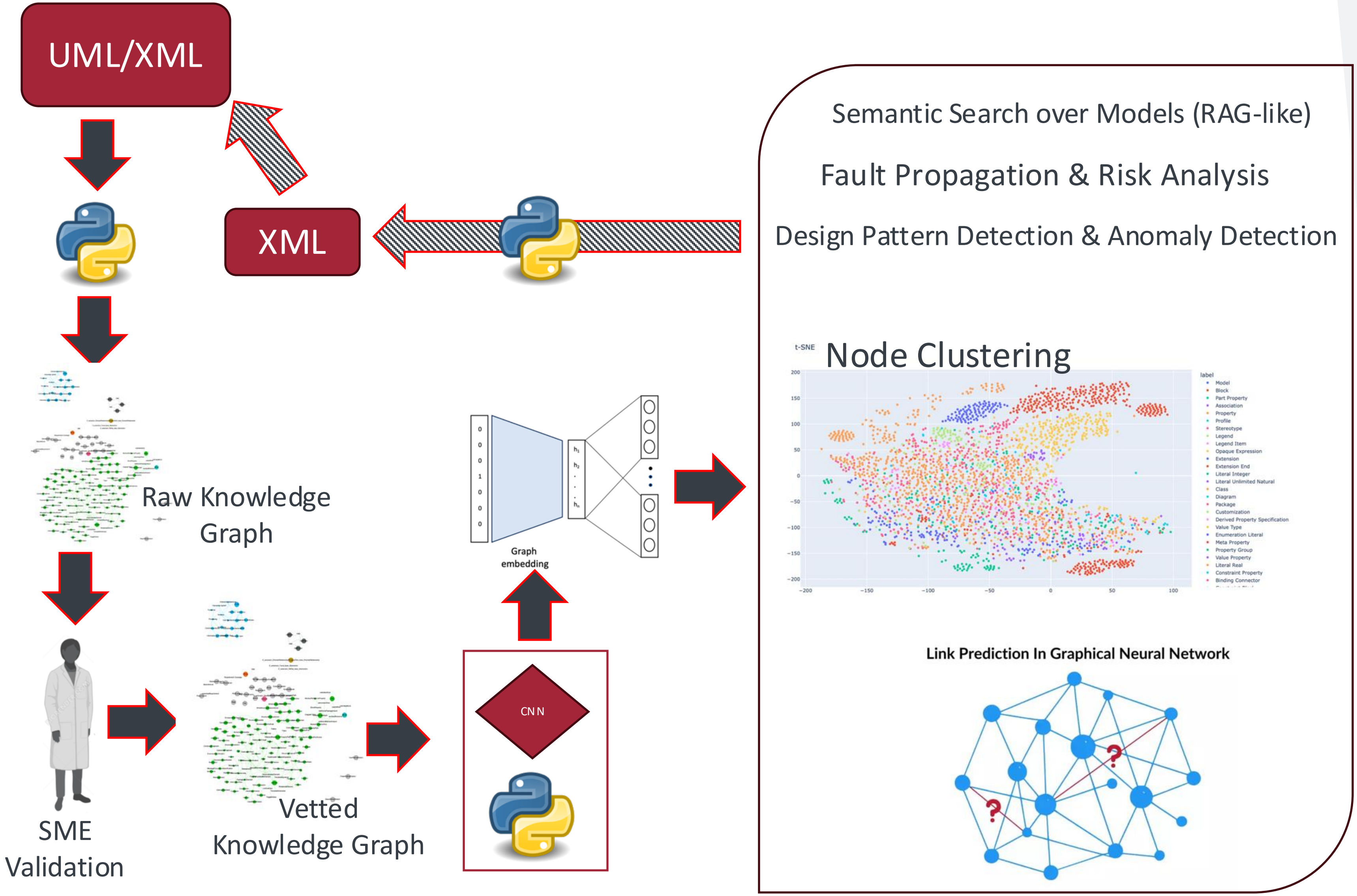
Prototype Case Study 2 – Using System's Specs - 2



Prototype Case Study 3 – Using a MBSE tool



Prototype Case Study 4 – Using a MBSE model



Outcomes and Impact

- The research significantly reduced manual effort and cognitive load for analysts reviewing and using complex system documentation
- It established practical methods for AI-assisted model generation that are compatible with MBSE and DE workflows
- By incorporating validation, GNN-based reasoning and requirement regeneration, the pipeline ensures trustworthiness and adaptability

Key Takeaways

- This research demonstrates a trustworthy, semi-automated pipelines that transforms legacy documentation into validated and adaptable system models
- The integration of GNNs introduces advanced reasoning capabilities and feedback mechanisms for continuously improving requirements
- By enabling regeneration of requirements in both English and SysML, the approach directly supports MBSE adoption
- Ultimately, this work provides a transparent, scalable and verifiable path for integrating AI into critical engineering practices

THANK ***YOU***

Stevens Institute of Technology
1 Castle Point Terrace, Hoboken, NJ 07030
clipizzi@stevens.edu