SYSTEMS ENGINEERING RESEARCH CENTER

CENTER FOR COMPLEX SYSTEMS & ENTERPRISES

Natural Language Processing (NLP) for Requirements Engineering

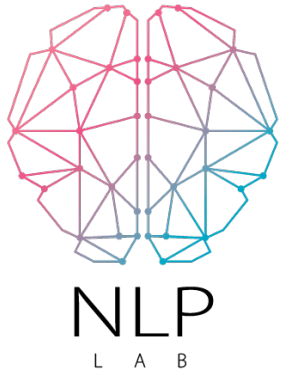
Extracting Formal Structures from Text

by
Dr. Carlo Lipizzi
clipizzi@stevens.edu

October 2021

Agenda

- The context
- NLP - NLU
- Why NLP/NLU for Requirements Extraction
- State of the art
- The Semantic Network Approach
- Current Developments
- Next Steps
- Q&A



The context

- A requirement is a singular documented need—what a particular product or service should be or how it should perform. It is a statement that identifies a necessary attribute, capability, characteristic, or quality of a system in order for it to have value and utility to a user [Mitre]
- Requirements engineering (RE) is the process of defining, documenting, and maintaining requirements in the engineering design process. It is a common role in systems engineering and software engineering [Wikipedia]

The context

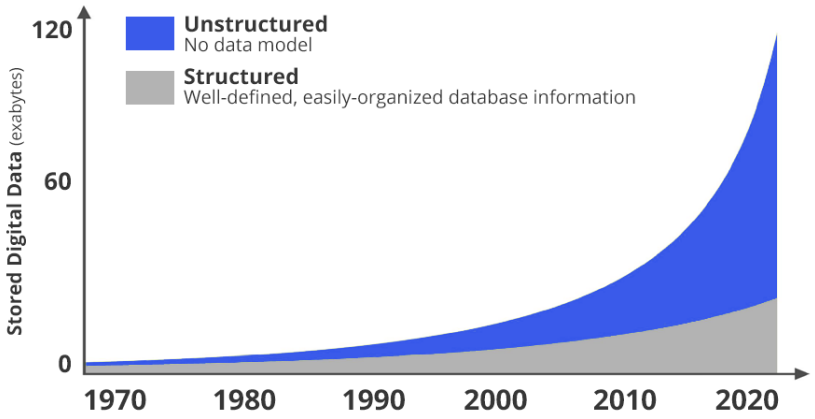
- A complete collection of requirements for a given system, can provide an abstract representation of the system itself. The resulting model is as accurate as the the process and the method that is used to generate it, within a given range of time validity
- The collection of requirements has traditionally been a top-down approach, requiring SMEs with a convergent vision
- SMEs may not be as available as needed, systems may change in time

The context

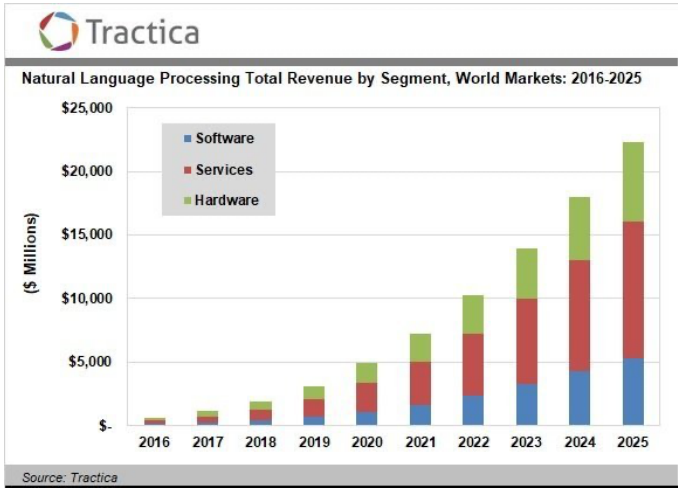
- We are focusing on requirement engineering with a “reverse engineering” approach, extracting “requirements” from existing material
- “Requirements” is a generic term and it may have different meaning, depending on the context
- It could be an ERA model, if the focus is on data representation, it could be a systemigram, if focus is on modeling a system, it could be a causal chain

Natural Language as source of Data

- 85-90 percent of all corporate data is in some kind of unstructured form, such as text and multimedia *[Gartner, 2019]*
- Tapping into these information sources is a need to stay competitive



Source: m-files.com



- Examples of application of **Natural Language Processing**: insurance (claim processing); law (court orders); academic research (research articles); finance (reports analysis); medicine (discharge summaries); technology (patent files); marketing (customer comments)

Challenges in Natural Language Processing

- Semantic ambiguity and context sensitivity
  - —automobile = car = vehicle = Toyota
  - —Apple (the company) or apple (the fruit)
- Syntactic/formal ambiguity
  - —Misspelling
  - —Different words for the same concept (e.g.: street; st.)
- Implicit knowledge
  - —We talk about things giving for granted common or specific knowledge

Implementing NLP

- Language is changing constantly, and NLP is following the changes, going from processing based on predefined structures (taxonomies/ontologies, syntax) to structures deducted from the text itself

Limitations of the traditional-deductive-"symbolic" approach

- Today, language is more fragmented, has less structure, has more jargons
- Different points of view may provide different interpretations

Machine Learning/inductive approach

- Extracting a numerical structure from text
- Different structures for different points of view
- Different structures automatically extracted over time

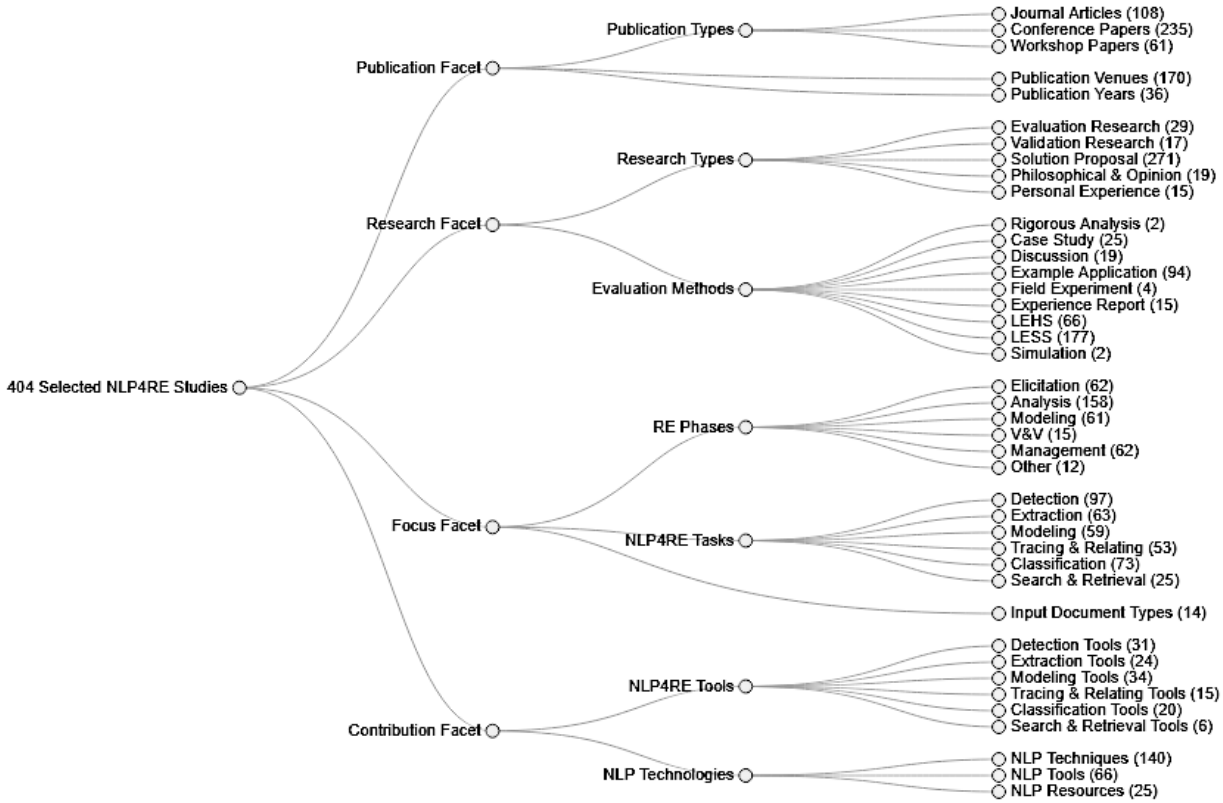
SYSTEMS ENGINEERING RESEARCH CENTER

Implementing NLP - limitations

- Understanding Language is not “just” processing. Understanding is a human characteristic, analyzed by philosophers as part of Epistemology
- An accurate (by human standard) “understanding” can come only from a model of human mind
- The current leading models in NLP/”NLU” are focused on the algorithmic part, missing a real model representing how the knowledge is created and used. It is basically representing the brain, not the mind. The leading model for NLP (GPT-3 by Open-AI) has 175 billion parameters, feeding a neural network providing results as a black box

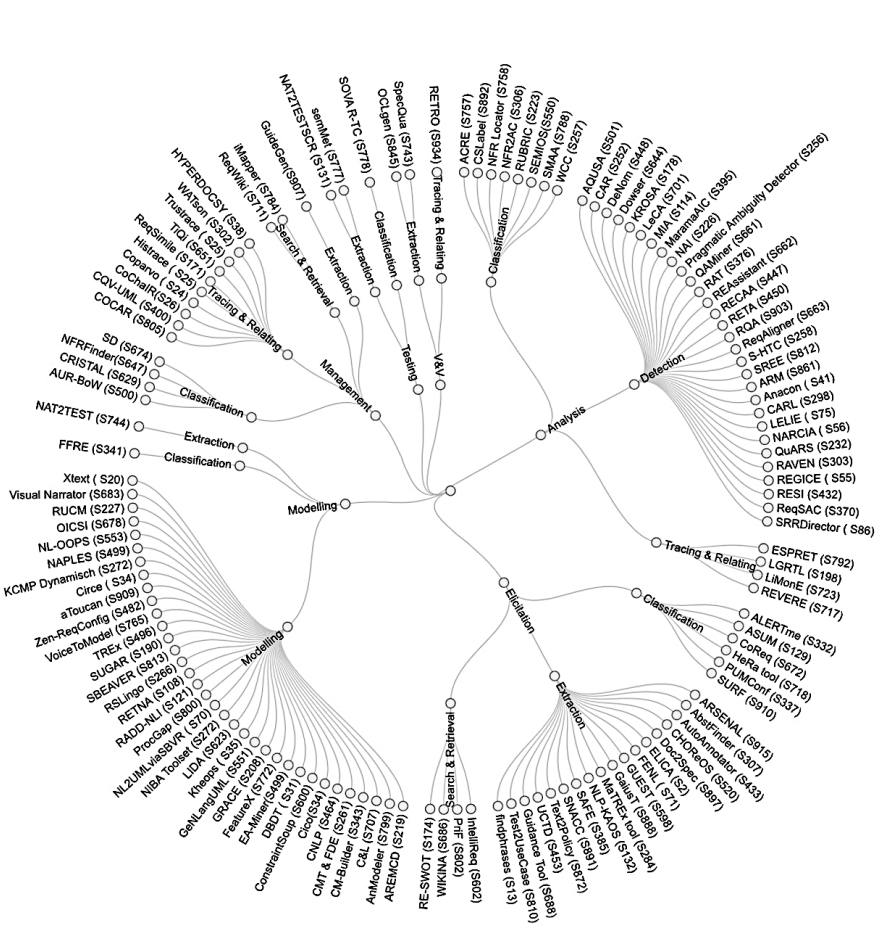
SYSTEMS ENGINEERING RESEARCH CENTER

Taxonomy and count of existing work on NLP for RE



L. Zhao et al., "Natural Language Processing for Requirements Engineering: A Systematic Mapping Study," ACM Comput. Surv., vol. 54, no. 3

NLP-based tools and techniques for RE



L. Zhao et al., "Natural Language Processing for Requirements Engineering: A Systematic Mapping Study," ACM Comput. Surv., vol. 54, no. 3

State of the art – bottom line

- We* considered 134 tools and approaches to apply NLP to Requirement Engineering
- The main insights resulting from the analysis were that no approach completely fulfilled the criteria of self extracting requirements/structures
- Solutions leveraging on Machine Learning with a semi supervised approach seems to be promising

M. Vierlboeck, C. Lipizzi, and R. Nilchiani - "Natural Language Processing for Requirements Engineering & Structure Extraction: An Integrative Literature Review" – Under review

Our Approach – 1st ingredient: the Network

- The approach is based on a corpus representing the domain we want to model
- The corpus does not have a formal structure connecting its semantic elements
- Using approaches based on words/n-grams proximity and applying techniques such as Word2Vec we create a semantic network representing the corpus
- In the network, the nodes are words/n-grams and the edges are calculated based on their proximity

Our Approach – 2nd ingredient: Network Theory

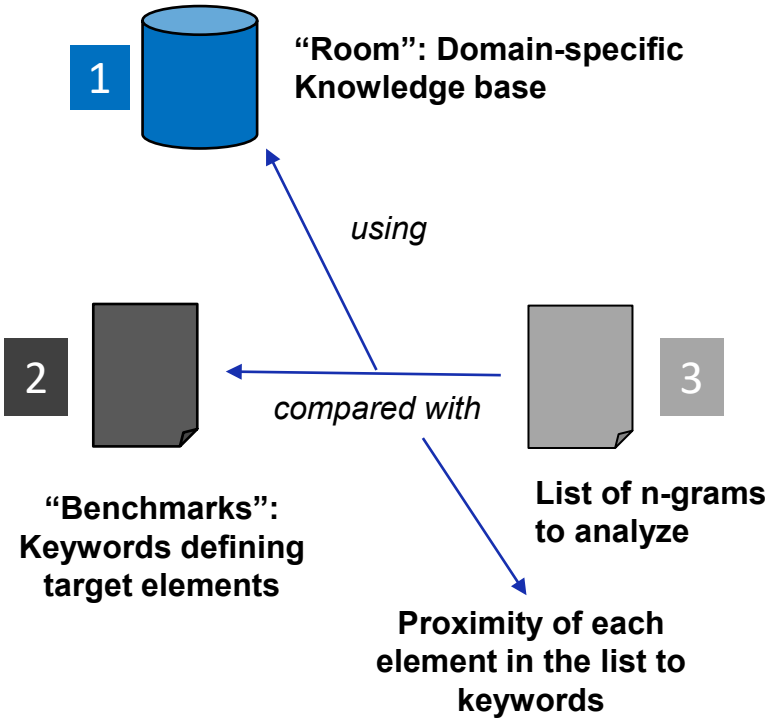
- On the network
  - We apply a partition/clustering method (based on Louvain Community Detection), creating “topics”
  - For the nodes in the cluster, we calculate a composite metric based on degree centrality, page rank and betweenness centrality
  - We pick the nodes with the highest values for the metric: those are the candidates “subjects” in their clusters

Our Approach – 3rd ingredient: The "Room" Theory*

- The "room theory" is a framework to address the relativity of the point of view by providing a computational representation of the context
- The non computational theory was first released as "schema theory" by Sir Frederic Bartlett (1886–1969) and revised for AI applications as "framework theory" by Marvin Minsky (mid '70)
- For instance, when we enter a physical room, we instantly know if it is a bedroom, a bathroom, or a living room
- Rooms/schemata/frameworks are mental frameworks we use to organize remembered information and represent an individuals/domain-specific view of the reality
- We create computational "rooms" by processing large corpora from the specific domain/community generating numerical dataset ("embeddings table"). The table is a representation of the words/ngrams, where each one of them is a n-dimensional vector and we use it as a knowledge base for the context/point of view

C. Lipizzi, D. Borrelli, and F. Capela, "The "Room Theory": a computational model to account subjectivity into Natural Language Processing

How the “room theory” works



- “Room theory” enables the use of context-subjectivity in the analysis of the incoming documents
- Context-subjectivity can be the point of of view of a subject matter expert
- The context-subjectivity in the analysis is represented by a domain specific numerical knowledge base, created from a large domain specific & representative corpus that is then transformed into a numerical dataset (“embeddings table”)

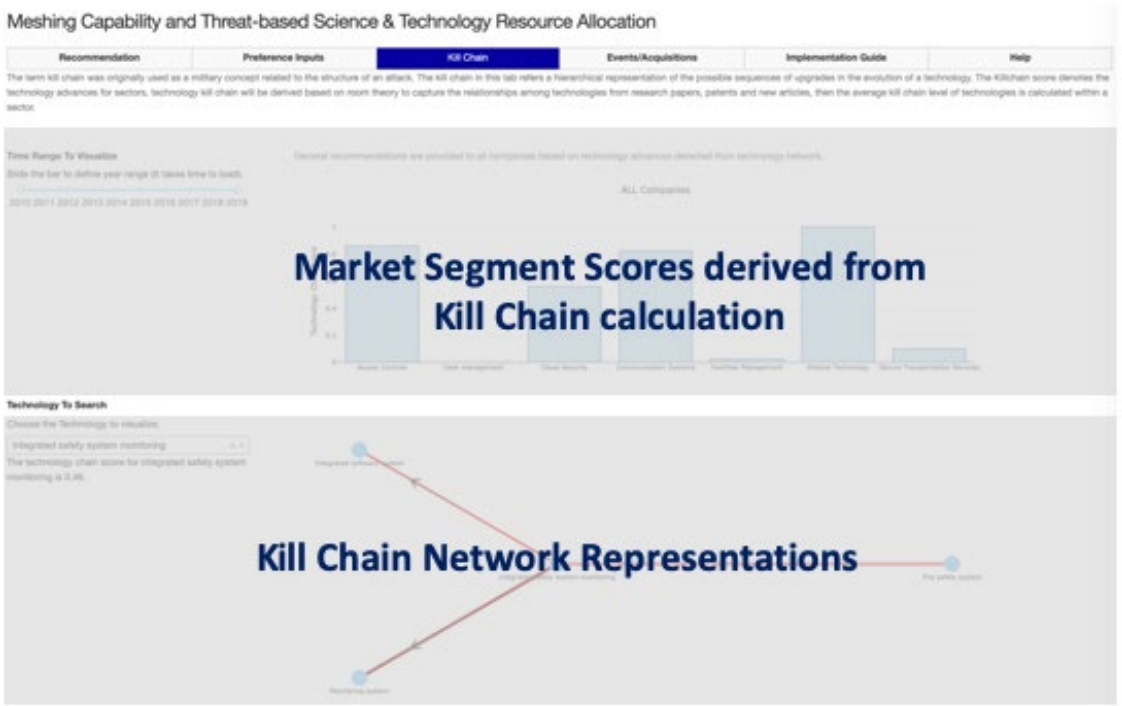
- The key components are:
  1. A point of view for the comparison (the “room”). This is represented by the embeddings table extracted from a large/representative corpus from the specific domain
  2. A list of “extended” keywords (using synonyms and misspellings) to be used for the analysis (the ”benchmark”)

Our Approach – putting things together

- We prune the list of ngrams using the room theory
- We create ego networks for the “subjects”. The degrees of separation is function of the size of the cluster
- The ego networks represent the semantic dependency between the nodes within the topics
- The approach can be extended to inter-clusters relations to recreate the complete formal representation
- Why all of this is relevant? The current ML-based models are limited to “similarity” between semantical elements, but they do not consider more complex relationships between them, such as semantic hierarchy

How we use it so far

- We used it to determine the causal chain in the domain of technologies
- Each technology has “components”, that are other technologies required for the first one. For example, cell. phones <- batteries, display, antennas, …



- The model has been partially implemented in WRT-1010 “Meshing Capability and Threat-based Science & Technology Resource Allocation”

Our Approach – next steps

With the proper funding, we will implement the following missing elements

- **Upgrading** the causal chain application/upgrading the overall approach:
  - Using the "room theory" to make the entities/nodes more relevant
  - Implement the "inter-clusters" relations
  - Implement a feedback mechanism to update the benchmarks
  - Test it on multiple domains
- **Extending** it to applications where edges/relationships have a semantic value, such as for Systemigrams and ERA
  - We will create a bipartite/2-mode graph *G* = (*E*, *R*, *A*) such that if e_i is an entity and r_j is a relationship, there is an edge $a_{ij} = (e_i, r_j) \in A$ if and only if e_i is associated to a relationship r_j
  - We will then extract a 1-mode graph with entities only. Two entities will be connected if and only if the share the same relationship. The common relationship will be the label in the Systemigrams or ERA
  - The extraction of relationships will be done using the "room theory"

STEVENS
INSTITUTE of TECHNOLOGY
THE INNOVATION UNIVERSITY®

Thank you!

Dr. Carlo Lipizzi
clipizzi@stevens.edu