UNIVERSITÀ DEGLI STUDI DI SALERNO
DIPARTIMENTO DI INFORMATICA

Q-DELOREAN

engineering of a QNLP pipeline for software requirements classification

Marco Calenda - 0522501165 **Prof. Fabio Palomba**

Software Engineering for Artificial Intelligence – 2022/2023

Introduction: Compositional and Distributional semantics

Compositional semantics

- Focuses on how the meaning of a complex expression is derived from the meaning of its constituents.
- Less success in real application.
- Closer view to a real human cognitive process, steps towards a *grammar-informed* NLP.

Distributional semantics

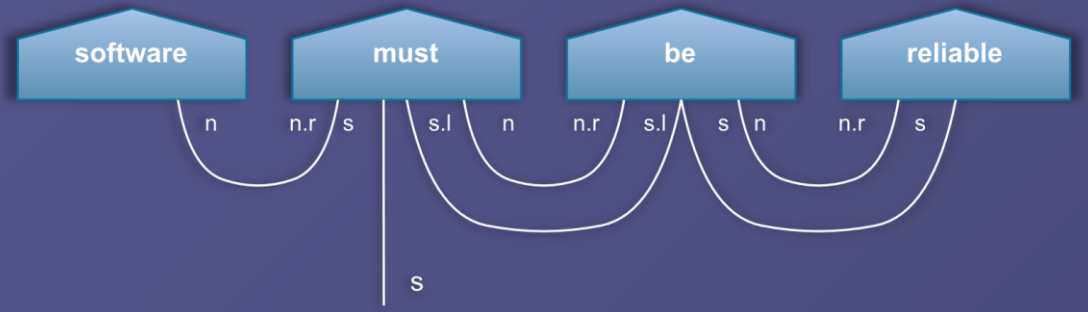
- Examines the relationships between words based on their co-occurrence patterns in a Corpus.
- Exploited by Big Data, is the current pivot of *state-of-the-art* NLP solutions (GPT-4, BERT).

Introduction: Compositional and Distributional semantics

Compositional semantics

- Focuses on how the meaning of a complex expression is derived from the meaning of its constituents.
- Less success in real application.
- Closer view to a real human cognitive process, steps towards a *grammar-informed* NLP.

- The *de facto* compositional framework used to model natural language is the model of Coecke et al. often dubbed as **DisCoCat**. It allows encoding sentences as **string diagrams** and **monoidal categories**:



- The underpinning mathematical are suitable for a translation into **quantum circuits** able to exponentially speed up the process (Quantum Natural Language Processing, QNLP).

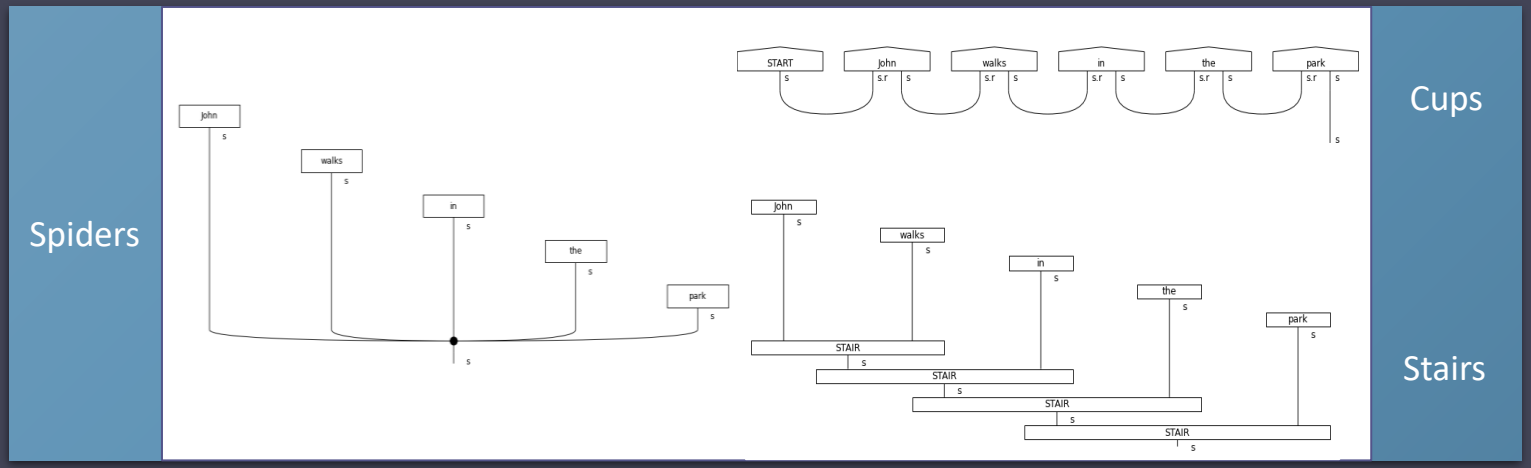
Introduction: Requirements classification with QNLP

- Individuation of **Non-Functional Requirements (NFRs):**
  - Time-consuming and experienced task.
  - They are not always explicit and concealed into ambiguous or imprecise sentences.
  - The lack of knowledge of NFRs in the early stages has a huge impact on the total cost and the failure rate of IT projects.

- By applying a so-called **quantum-inspired** approach, namely a solution trained and tested on classical hardware yet quantum-ready, we explore the QNLP potential applied into a quasi-real scenario task of binary classification of FRs/NFRs.

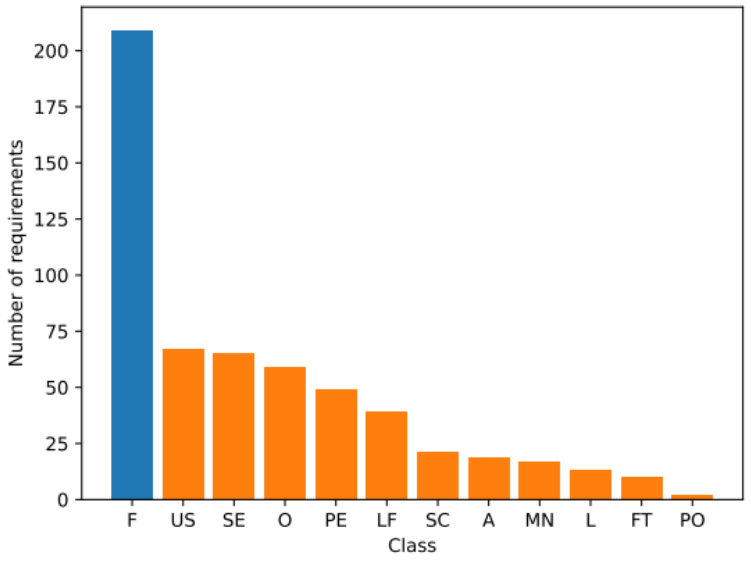
Research Questions

- **RQ1.** To what extent can the proposed tool classify FRs and NFRs?
- **RQ2.** How does the **DisCoCat** framework compared to other compositional models?



- **RQ3.** How does the proposed tool compared to classical NLP models?
  - **TF-IDF** & Naïve Bayes
  - **Word2Vec** & Feed-Forwarded Neural Network

PROMISE NFR dataset



15 Projects

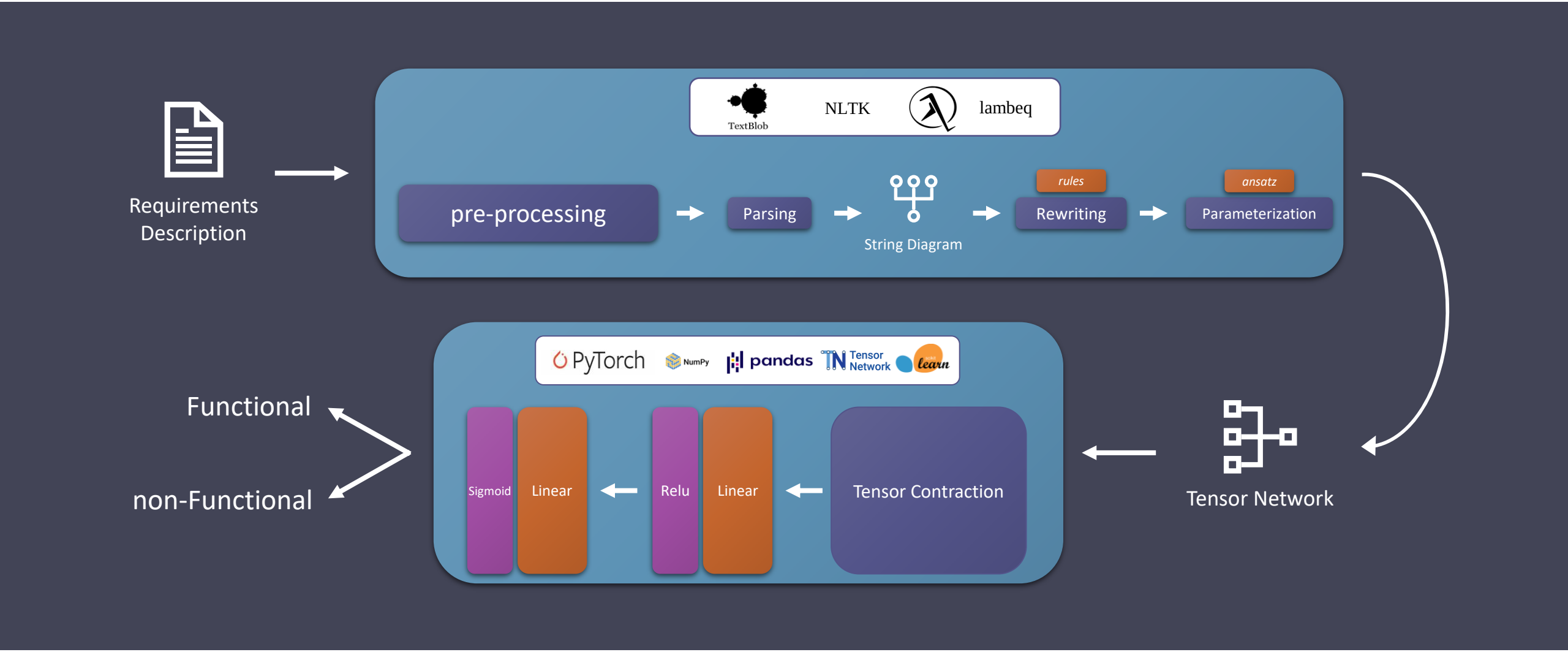
255 FRs

370 NFRs

- **Manual Filtering**
  - Large requirements description
  - Sentences with a complex grammatical structure
- **Data Balancing with *Undersampling***

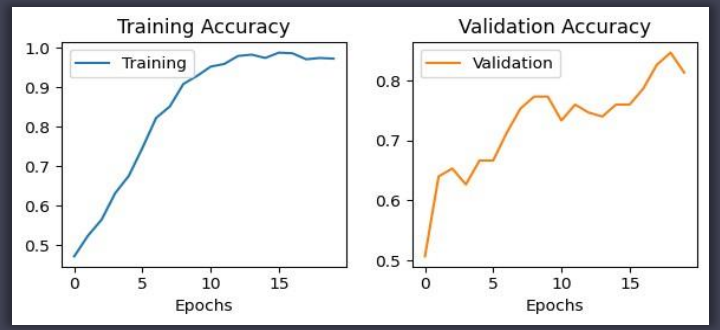
Data Collection

Q-DELOREAN


Requirements Description
TextBlob
NLTK
lambeq
pre-processing
Parsing
String Diagram
rules
Rewriting
ansatz
Parameterization
Tensor Network
PyTorch
NumPy
pandas
Tensor Network
learn
Sigmoid
Linear
Relu
Linear
Tensor Contraction
Functional
non-Functional

Grid Search & Cross-Validation

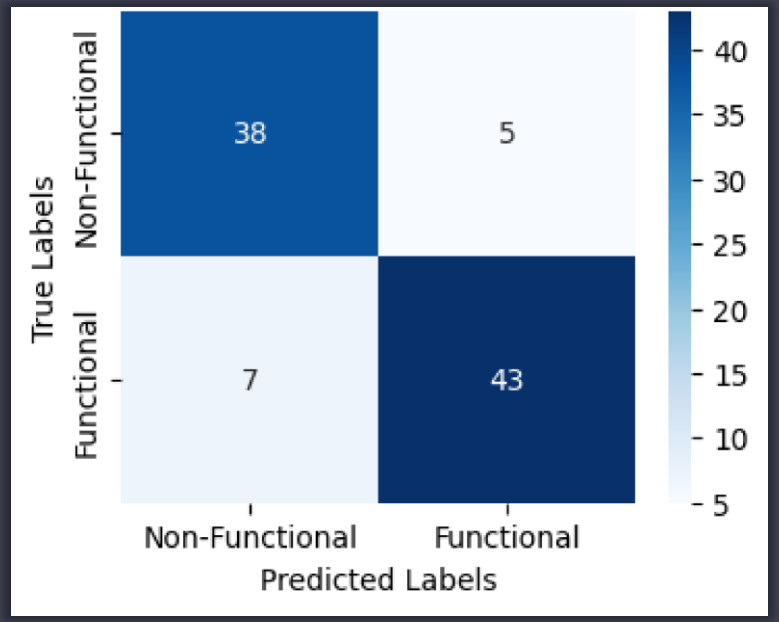
Learning Rate	Batch Size	Hidden Layers Size	Training Accuracy	Validation Accuracy
0.07	32	32	90.9%	81.3%
0.07	**32**	**8**	**97.4%**	**86.0%**
0.03	16	16	87.1%	79.2%
0.1	16	16	78.2%	70.8%



The experiments are conducted on the *Google Colab Pro* platform using a hardware accelerated runtime with a GPU NVIDIA A100 Tensor Core.

Results: Readers Comparison

	Reader	Training Accuracy	Test Accuracy
4-Dim	Stairs	80.0%	68.8%
	Cups	84.9%	76.3%
	Tree	90.2%	83.9%
	DisCoCat	**97.4%**	**87.6%**
2-Dim	Spiders	93.8%	69.2%



The experiments are conducted on the *Google Colab Pro* platform using a hardware accelerated runtime with a GPU NVIDIA A100 Tensor Core.

Results: Models Comparison - 1

Model	No. of parameters	Test Accuracy	Training time (15 epochs)	Inference time (100 sentences)
Q-DELOREAN	**58**	87.6%	**1.3 min**	5 s
TF-IDF Naïve Bayes	0	75.2%	-	-
Word2Vec NN	**23k**	88.2%	30 s	2 s

The experiments are conducted on the *Google Colab Pro* platform using a hardware accelerated runtime with a GPU NVIDIA A100 Tensor Core.

Results: Models Comparison - 2

Model	Emissions ($CO_2\ kg$)	GPU Energy Consumed (kW)	RAM Energy Consumed (kW)	Total Energy (CPU+GPU+RAM)
Q-DELOREAN	**2.35 e+12**	5.79 e+14	2.03 e+14	**16.94 e+14**
TF-IDF Naïve Bayes	1.02 e+11	1.78 e+9	9.04 e+7	**5.27 e+7**
Word2Vec NN	1.84 e+12	1.61 e+14	**9.17 e+14**	3.77 e+14

The experiments are conducted on the *Google Colab Pro* platform using a hardware accelerated runtime with a GPU NVIDIA A100 Tensor Core.

Conclusions and future works

- Our study demonstrates the effectiveness of Q-DELOREAN in classifying requirements obtaining overall promising results.
- The findings of this study open up several avenues for future research:
  - Further exploration can be done to enhance the performance of the Linear Readers by incorporating additional linguistic features and/or ad-hoc classifiers.
  - Extending this approach to other types of languages, e.g. programming languages, could prove to be effective since their fixed and limited grammar makes them well-suited for compositional modeling techniques like the one employed in our study.

UNIVERSITÀ DEGLI STUDI DI SALERNO
DIPARTIMENTO DI INFORMATICA

Thanks for the attention!