

TooDY

a Tool for Detecting variabilityY

Matteo Giorgi

Università degli Studi di Pisa

01 Dicembre 2023



Scopo del Progetto

Realizzare uno strumento di elaborazione del linguaggio naturale, per individuare indicatori di variabilità in un documento dei requisiti.

Il progetto implementa una web-app **Flask**, che esegue analisi lessicale e sintattica, con il supporto della libreria open-source **spaCy**.

SPL Engineering

È un paradigma di sviluppo software che modella le caratteristiche comuni e quelle di variabilità di una famiglia di sistemi correlati, al fine di massimizzare il riutilizzo e ridurre i costi di sviluppo.

Variabilità in un requisito

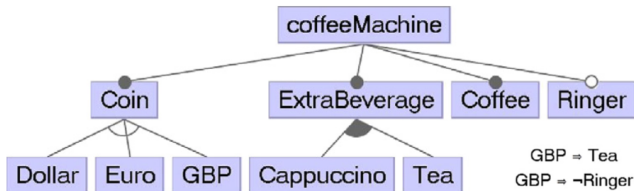
Rappresenta la capacità del requisito stesso di corrispondere a più di una possibile configurazione e di essere modificato o esteso a seconda del contesto specifico in cui viene implementato.

I **difetti di ambiguità** nei requisiti possono fornire un'indicazione di variabilità (in design, implementazione o configurabilità).

- **Vaghezza** – parole con significato non quantificabile o univoco.
- **Debolezza** – presenza di un verbo debole (non imperativo).
- **Forma passiva** – ambigua se non è chiaro chi compie l'azione.
- **Opzionalità** – presenza di parole come *possibly* o *probably*.
- **Molteplicità** – riferimento a una pluralità di oggetti (*and*, *or*).
- **If, When, Where** – clausole subordinate che indicano condizioni.

Caso di Studio: Coffee Vending-Machine

- R1.** After inserting a **suitable** coin, the user shall choose a beverage and select the amount of sugar.
- R2.** The machine shall offer, as beverages, coffee **and** cappuccino **or** tea.
- R3.** The machine shall **always** offer coffee.
- R4.** A ringtone **possibly** has to **be played** after beverage delivery.
- R5.** After the beverage **is taken**, the machine returns idle.
- R6.** The British market requires tea and excludes **any** ring tone.



Dato un testo, l'**elaborazione del linguaggio naturale** prevede quattro fasi distinte per analizzarlo e comprenderlo.

- 1 **Tokenizzazione** – Suddivide il testo in unità significative (*token*).
- 2 **An.Morfologica** – Identifica il ruolo dei *token* nel discorso.
- 3 **An.Sintattica** – Studia la struttura grammaticale del periodo.
- 4 **An.Semantica** – Interpreta il significato delle strutture sintattiche.

Cos'è spaCy?

È una libreria Python open-source per NLP, che offre parsing sintattico e semantico, tokenizzazione, PoS-Tagging, Named Entity Recognition, supporta modelli di apprendimento automatico e integrazione con TensorFlow e PyTorch.

Funzionalità di spaCy

- ① Tokenizzazione
- ② PoS-Tagging
- ③ Dependency Parsing
- ④ Lemmatizzazione
- ① Sentence Boundary Detection
- ② Named Entity Recognition
- ③ Rule-Based Matching

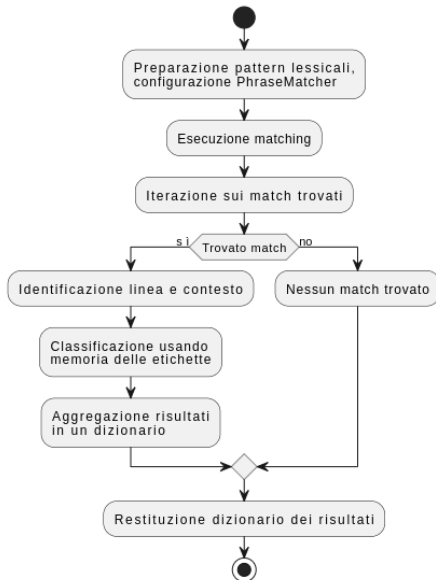
Esempio di Rule-Based Matching con Matcher

```
import spacy
from spacy.matcher import Matcher

nlp = spacy.load("it_core_news_sm")
matcher = Matcher(nlp.vocab)
pattern = [{"POS": "NOUN"}, {"POS": "ADJ"}]
matcher.add("NOUN_ADJ_PATTERN", [pattern])

doc = nlp("Il mercato azionario ha mostrato una crescita costante.")
matches = matcher(doc)
```

Parser lessicale



Parser forma passiva

```
import spacy
from spacy.matcher import Matcher

nlp = spacy.load("it_core_news_sm")
matcher = Matcher(nlp.vocab)
pattern = [{"POS": "NOUN"}, {"POS": "ADJ"}]
matcher.add("NOUN_ADJ_PATTERN", [pattern])

doc = nlp("Il mercato azionario ha mostrato una crescita costante.")
matches = matcher(doc)
```


Parser di congiunzioni verbali

```
import spacy
from spacy.matcher import Matcher

nlp = spacy.load("it_core_news_sm")
matcher = Matcher(nlp.vocab)
pattern = [{"POS": "NOUN"}, {"POS": "ADJ"}]
matcher.add("NOUN_ADJ_PATTERN", [pattern])

doc = nlp("Il mercato azionario ha mostrato una crescita costante.")
matches = matcher(doc)
```

Analizzatore di condizioni

```
import spacy
from spacy.matcher import Matcher

nlp = spacy.load("it_core_news_sm")
matcher = Matcher(nlp.vocab)
pattern = [{"POS": "NOUN"}, {"POS": "ADJ"}]
matcher.add("NOUN_ADJ_PATTERN", [pattern])

doc = nlp("Il mercato azionario ha mostrato una crescita costante.")
matches = matcher(doc)
```