

# Report - NLU Assignment 1

Davide Lusuardi

April 19, 2021

# 1 Description

In this report we present some implementation choices of the first assignment of the course Natural Language Understanding.

The following functions has been implemented.

- **extract\_path(sentence)**. This function extracts for each token a path of dependency relations from the ROOT to the token. We first parse the sentence to get a Doc object of spaCy and then, for each token in the sentence, we find the path of dependency relations from the ROOT to the token.

The function accepts a sentence of type **string** and returns a dictionary containing for each token (keys in the dictionary) the list of dependency relations.

- **extract\_subtree(sentence)**. This function extracts the subtree of each token in the sentence. We first parse the sentence to get a Doc object of spaCy and then, for each token, we extract the subtree using the attribute **Token.subtree**.

The function accepts a sentence of type **string** and returns a dictionary containing for each token (keys in the dictionary) a list of tokens, ordered with respect to the sentence, representing the subtree.

- **check\_subtree(sentence, words)**. This function checks if the given list of words forms a subtree of the dependency graph of the sentence.

The function accepts a sentence of type **string** and a list of words (list of **string**). It returns a boolean value, **True** if the words form a subtree.

The function extracts all the subtrees of the sentence using the function **extract\_subtree()** and checks if the list of words match a subtree.

- **get\_head(span)**. This function gets the head of the given span. From spaCy documentation, the head token is such that **token.head == token**.

The function accepts a span of type **string** and returns a token of type **spacy.tokens.token.Token**.

- **extract\_nsubj\_dobj\_iobj(sentence)**.

This function extracts the sentence subject (nsubj), direct object (dobj) and indirect object (iobj) spans from the sentence.

The function accepts a sentence of type **string** and returns a dictionary containing "nsubj", "dobj", "iobj" as keys and the relative list of spans as values. If for example there are more subjects in the sentence, the entry "nsubj" will be a list containing all the subject spans. A span is represented as a list of tokens.