Nomes: Douglas Costa, Jader Fróes, Patrick Guimarães

Extração de Relação

Extração:
Para extrair a frase onde está contida a relação, é extraído inicialmente argumento 1 e sua categoria e argumento 2 e sua categoria.
Utilizando a função abaixo, retiramos a sentença que está contida entre o argumento 1 e argumento 2. Dentro desta sentença temos a relação.

```
for i in range(len(df)):
  text = df["SENTENCE"][i]
  start = df["ARGUMENT_1"][i]
  end = df["ARGUMENT_2"][i]
  rel = text[text.find(start)+len(start):text.rfind(end)]
  rel = normalize(rel)
```

Normalização:
Para extrair as relações, as sentenças foram normalizadas através de uma função que substitui algumas palavras do dataset por outras corretamente escritas, usando a biblioteca ‘re’. Exemplos:

```
    text = re.sub(" de a "," da ",text)
    text = re.sub(" de o "," do ",text)
    text = re.sub(" de os "," dos ",text)
    text = re.sub(" de as "," das ",text)
    text = re.sub(" em a "," na ",text)
    text = re.sub(" em o "," no ",text)
    text = re.sub(" em os "," nos ",text)
    text = re.sub(" em as ", "nas ",text)
    text = re.sub("á","a",text)
    text = re.sub(" por a "," pela ",text)
    text = re.sub(" por o "," pelo ",text)
    text = re.sub(" por os "," pelos ",text)
    text = re.sub(" por as "," pelas ",text)
    text = re.sub("\((?! )","( ",text)
    text = re.sub("\r","",text)
    text = re.sub(",","",text)
```

Nesse caso a função substitui a primeira string pela segunda.

Categorização:
Foi utilizado a biblioteca spacy que possui suporte para o português. Spacy utiliza rede neural para categorizar o part-of-speech.

```
doc = nlp(u'estão a ignorar as leis da')
[(token.orth_, token.pos_) for token in doc]
[('estão', 'AUX'), ('a', 'ADP'), ('ignorar', 'VERB'), ('as', 'DET'),
('leis', 'NOUN'), ('da', 'ADJ')]
```

Regras:

ReVerb extrai relacionamentos com base em uma restrição simples, toda frase relacional, ou seja, a sequência de palavras que conecta duas entidades, deve ser:
um verbo , um verbo seguido imediatamente por uma preposição ou um verbo seguido de substantivos, adjetivos ou advérbios que terminam em uma preposição

Foi utilizado o RegexpParser do NLTK para codificar a expressão regular do ReVerb adaptada ao português.
As regras criadas para extração de relações foram:

```
verb = "<ADV>*<AUX>*<VERB><PART>*<ADV>*|<NOUN|PROPN>*"
word = "<NOUN|ADJ|ADV|DET|ADP>"
preposition = "<ADP|ADJ>"
rel_pattern = "( %s (%s* (%s)+ )? )+ " % (verb, word, preposition)
grammar_long = '''REL_PHRASE: {%s}''' % rel_pattern
reverb_pattern = nltk.RegexpParser(grammar_long)
```

Após a regra ser aplicada na sentença, a relação de maior tamanho é retornada como a relação das duas entidades.

Link para o código: https://github.com/patrickguima/PLN