ADFN, week 3 day 3: fact extraction

Facts are present in documents. They can be extracted in many ways. Two ways we saw in class are, by using regular expressions, and by using dependency parsing.

Facts were extracted, in this case, as <entity, predicate, value> triples; e.g.

> <Lesotho, has-population, 2 million>
> <Russia, has-capital, Moscow>

Reading:

> https://arxiv.org/abs/1811.05768
> http://eprints.whiterose.ac.uk/91378/1/Identification%20and%20verification.pdf
> https://www.aclweb.org/anthology/N18-1074.pdf

For this assignment, write a fact extraction method, or improve the one presented in the notebook in class. You might like to write your own regular expressions, for any language you like. Or, you could use the findSVOs method, but make it do better named entity recognition.

You should:

1. Write a triple extraction system. You can use regular expressions or dependency parsing. Use the shared Colab. You can also use spaCy or nltk or any other library.

   The system should extract entity names, e.g. “Russian Federation”, instead of just single words. Look at this for an example of how to recognise them: https://spacy.io/usage/rule-based-matching#models-rules-ner

   Describe how your approach works.
2. Decide on two different text sources.

   Try to make sure they have different styles or biases.
3. Look at the triples that you get.

   Give some examples, and talk about their quality.
4. Compare the triples with Wikidata content.

   What things do you find that are not in Wikidata?

   What is in Wikidata but not the automatically extracted triples?

Give a five-minute presentation on your system in the next class.