D5: Databases and Information Systems
Automated Knowledge Base Construction, SS 2022
Lab #06: Open Information Extraction

max planck institut
informatik

Problem 1 (Open IE).
task Description: In this lab, we are working on open information extraction: given a sentence, extract all statements as S-P-O (subject-predicate-object).

You can use any tool to pre-process the data, like POS tagging, dependency parsing, etc. You may also use pretrained word embeddings like word2vec or BERT. However, you are not allowed to use any existing open information extraction or semantic role labeling tools.

Data and Scripts: For this lab, we will use a dataset[1] with 1601 input sentences. Download `lab06.zip`, extract and **rename** the folder `Lab06_XXXXX_YourName` with *your* matriculation number and name. The lab data is comprised as follows:

- `oei_corpus/sentences.txt` : Sample input statements.
- `oei_corpus/extractions-groundtruth.oie` : corresponding GT statements. Each line represents a single Open IE extraction, in a tab separated format:

```
TOKENIZED_SENTENCE [tab] PREDICATE_HEAD [tab] FULL_PREDICATE [tab] ARG1 [tab] ARG2 [tab]....
```

- `oie_readers/oieReader.py` : Super class for defining Open IE readers.
- `oie_readers/goldReader.py` : Reader for GT Open IE statements.
- `oie_readers/clausieReader.py` : Reader for predicted Open IE statements.
- `oie_readers/extraction.py` : Used by reader functions to hold sentence, single predicate and corresponding aguments.
- `oie_readers/matcher.py` : Has different functions for comparing predictions with the GT.
- `benchmark.py` : Loads the GT file and compares predictions with the GT using a lexical matcher.
- `run.py`: takes `./oie_corpus/sentences.txt`) as the input and stores extractions in `results.txt`.
- `run_evaluate.sh` : Calls `benchmark.py` which evaluates `results.txt` against the GT by computing the precision and recall (P/R) scores which are saved in `pr_result.dat`.

The output file (`results.txt`) has the following format:

```
sentence_1
id_of_sentence_1 [tab] "subject_1" [tab] "predicate_1" [tab] "object_1" [tab] 0
id_of_sentence_1 [tab] "subject_2" [tab] "predicate_2" [tab] "object_2" [tab] 0
...
```

The last column represents the confidence score, **which should always be** `0`.

Baseline: The `run.py` has a baseline (F1 score: 0.14). This baseline method extracts all triples of nominal subject, verb, direct object from sentences. The result is in `baseline_results.txt`.

You can run and evaluate your program using: `./run_evaluate.sh`. `./run_evaluate.sh` first executes `run.py` to extract SPO triples from `oie_corpus/sentences.txt` and saves them to `results.txt`. It then executes `benchmark.py` to evaluate P/R scores and saves them in `pr_result.dat` file.

Submission: Your submitted files must include the main program file `run.py` and any other file/code that you have additionally used. If you used any external libraries, please indicate them in a README file. Your file **need not** contain `oie_readers/`, `oie_corpus/`, `benchmark.py`, `run_evaluate.py`

Please submit all necessary files, which are compressed into a zip file named:
Lab06_MatriculationNumber_Name.zip
to the email address: **akbc-assignments@mpi-inf.mpg.de** with title of the email: **[AKBC]Lab06_MatriculationNumber_Name**

Deadline: 23:59 07.06.2022 (Tuesday)

[1] Taken from Stanovsky & Dagan: Creating a Large Benchmark for Open Information Extraction, *EMNLP 2016*.